APPENDIX

Design Document

for ZendIt

Zendit 1959 NW Dock, Seattle Washington 98107 206.783.8000

Table of Contents

1. Introduction

1.1 Purpose

Zendit offers services to enhance online personal privacy. Aditi has proposed to build a system that will enable users to encrypt and decrypt online messages, manage keys, manage the storage of users' personal information and manage the health and welfare of the entire system.

This document provides an overview of the high-level design of the entire system.

1.2 Scope

The document would provide an insight into the implementation of the pages in the web site and the SurfBoard components. This document would also provide details regarding the interaction between the SurfBoard and the web site for data requests.

1.3 Target audience

This document is intended for the engineers responsible for developing and testing the Zendit system and the concerned Zendit system administrators.

1.4 Organization

The design has been explained under three main sections:

1. The Zendit Web Site Design
2. The SurfBoard Design
3. The Database Design

The other details that are covered in the document are
a. Implementation of the hashing algorithm
b. Registration and Key confirmation process
c. Security on the pages.

1.5 Software for implementation

- ASP pages will be used on the web site to process all server requests.
- SQL Server 7.0 will be used as the database server.
- The SurfBoard components would be implemented as COM objects using ATL
- The wrapper for OpenLib will be built using C/C++

2. Database Design

This section presents the details of the database design and gives the description of all the tables in the system.

Table Name: tblRegisteredUsers
Table Description:
This table would contain all the personal information about the user and certain other information related to the Zendit activities.

Field Name	Field Type	Length	Field Description
LoginID	varchar	20	Login Id given by the user. The login Id would be unique for each user.
Password	varchar	20	Password given by the user.
PrimaryEmailID	varchar	250	This is the email ID that would be used for all communications between Zendit system and the user.
FirstName	varchar	100	First name of the user.
LastName	varchar	100	Last name of the user.
DateOfRegistration	datetime	8	Date on which the user registers with Zendit system.
Address1	varchar	300	Address of the user.
Address2	varchar	300	Address of the user.
City	Varchar	50	Name of the City
State	Varchar	50	Name of the State

Field Name	Field Type	Length	Field Description
Country	Varchar	50	Name of the Country
ZipCode	Varchar	50	The zip code of the user
ResidencePhone	varchar	25	Residence phone number.
OfficePhone	varchar	25	Office phone number.
Mobile	varchar	25	Mobile phone number.
Fax	varchar	25	Fax number.
ConfirmationFlag	int	4	A flag indicating whether the user has confirmed the registration or not. The flag is set to "True" on confirmation of registration.
DateOfConfirmation	datetime	8	Date on which the user confirms the registration.
NewMailStatus	int	2	A flag indicating whether the user has any new mails. It is set to "True", if any new mail is received by the user and is set to "False", if any of the mails (new or old) is read by the user.
LastLoginDate	datetime	8	The date on which the user last logged into Zendit system.
TotalSpaceAlloted	float	8	Indicates the total free hard disk space given to the user by the Zendit server.

Field Name	Field Type	Length	Field Description
FreeSpaceLeft	float	8	Indicates the total unused hard disk space left for the user.
ListOfEmailID	varchar	2520	Contains a string of email IDs (those in the certificates associated with the user's login ID) separated by commas.

Table Name: tblRegistrationConfirmation
Table Description:
This table would contain confirmation details for every successful registration.

Field Name	Field Type	Length	Field Description
ConfirmationID	varchar	20	Confirmation ID (for registration) sent to the user.
LoginID	varchar	20	Login ID of the user.
DateOfRegistration	datetime	8	Date on which the user registers with Zendit system.
iMailSent	Bit	1	Shows status of Confirmation Mail (0 if mail not sent and 1 otherwise)

Table Name: tblNonConfirmedCerts
Table Description:
This table would contain non-confirmed certificates.

Field Name	Field Type	Length	Field Description
EmailID	varchar	250	Email ID specified in the certificate.
KeyID	varchar	17	Key ID specified in the certificate.
Name	varchar	250	Name specified in the certificate.
Certificate	text	16	The digital certificate.
MailSystem	varchar	50	Mail system specified in the email ID field of the Certificate.
InsertedbyZend	int	4	A flag indicating whether the certificate was inserted by Zendit system. If so, flag is set to "True".
vAlgo	Varchar	250	The Algorithm used for creating the Key pair
vLength	Varchar	250	The length of the key generated
vType	Varchar	250	The Key Type which is stored. (As of now only Public keys are stored in this table)
dtCreate	Datetime	8	Date of Key Creation

Field Name	Field Type	Length	Field Description
dtExpiry	Datetime	8	Date of key Expiry
MailSent	int	4	A flag indicating whether the confirmation mail has been sent to the user. If the mail is sent successfully then the flag is set to "True" else it is set to "False".

Table Name: tblKeyConfirmation
Table Description:
This table is an intermediate table that holds the login ID - key ID relation till the user confirms the key pair.

Field Name	Field Type	Length	Field Description
ConfirmationID	varchar	20	Confirmation ID (for using the key Id) sent to the user.
LoginID	varchar	20	Login ID of the user.
KeyID	varchar	17	Key ID as in the certificate.
DateOfCreation	datetime	8	Date on which the certificate was created.
EmailID	varchar	250	EmailID associated with the key
MailSent	int	4	Flag which tells if the mail has been sent. (–1 for no)

Table Name: tblConfirmedCerts
Table Description:
This table contains all the certificates associated with a login ID.

Field Name	Field Type	Length	Field Description
LoginID	varchar	20	Login ID of the user.
EmailID	varchar	250	Email ID as in the certificate.
Name	varchar	250	Name as in the certificate.
KeyID	varchar	17	Key ID as in the certificate.
MailSystem	varchar	50	Mail system specified in the email ID field of the certificate.
Certificate	text	16	The digital certificate of the user.
NumberOfMailsZend	int	4	The number of mails zent by the user using the email ID.
vAlgo	Varchar	250	The Algorithm used for creating the Key pair
vLength	Varchar	250	The length of the key generated
vType	Varchar	250	The Key Type which is stored. (As of now only Public keys are stored in this table)
dtCreate	Datetime	8	Date of Key Creation
dtExpiry	Datetime	8	Date of key Expiry

Table Name: tblBadCerts
Table Description:
This table would have all the certificates that have been:

- Moved from the table tblConfirmedCerts (tblBadCerts.Flag = Replaced)
- Moved from the table tblNonConfirmedCerts (tblBadCerts.Flag = Deleted)

Field Name	Field Type	Length	Field Description
EmailID	varchar	250	Email ID as in the certificate.
KeyID	varchar	17	Key ID as in the certificate.
Name	varchar	250	Name as in the certificate.
Flag	int	4	The flag Indicates whether the certificate was 'Replaced' or 'Deleted' .
Certificate	text	16	The digital certificate.

Table Name: tblKeyUpload
Table Description:
This table would contain all the locks and keys uploaded by the user.

Field Name	Field Type	Length	Field Description
LoginID	varchar	20	Login ID of the user.
EmailID	varchar	250	Email ID as in the lock and key uploaded by the user.
KeyID	varchar	17	Key ID as in the lock and key uploaded by the user.
Name	varchar	250	Name as in the lock and key uploaded by the user.

Field Name	Field Type	Length	Field Description
FileBlob	text	16	lock and key uploaded by the user stored in blob format.
DateOfUpload	datetime	8	Date on which the user uploaded the lock and key.

Table Name: tblFileUpload
Table Description:
This table would contain all the information about the files being uploaded by the user onto the Zendit server.

Field Name	Field Type	Length	Field Description
IFileListID	Int	4	List ID of File
LoginID	varchar	20	Login ID of the user.
NameOfFile	varchar	800	Name of the file uploaded by the user.
FileSize	float	8	Size of the file uploaded by the user.
DateOfUpload	datetime	8	Date on which the user uploaded the file.
FilePath	varchar	200	Path (the entire network path) to the file where the user uploads in the Zendit server.
ContentType	Varchar	500	

Note:- This Feature has been moved to later version

Table Name: tblUserProfile
Table Description:
This table would contain all the personal information of user that would be shared with other users.

Field Name	Field Type	Length	Field Description
LoginID	varchar	20	Login ID of the user.
ProfileString	varchar	5000	The user would like to share some of his/her personal information (first name, last name, phone number etc.) with the other users. This field would contain the respective column names (of tblRegisteredUsers table) separated by commas as a single string.

Table Name: tblAdminSettings
Table Description:
This table would contain all the admin settings for both the web site and SurfBoard set by an administrator.

Field Name	Field Type	Length	Field Description
Key	varchar	20	Key name
KeyDesc	varchar	200	Key description
Value	varchar	200	Key value

The administrator is expected to update settings such as:
Example:

1) KeyID = iMaxCerts
KeyDescription = Maximum certificates per user
Value = 10

2) KeyID = iTimeInactive
KeyDescription = Time limit for inactivity
Value = 60

Table Name: tblFeedback
Table Description:
This table would contain the feedback information given by any registered or unregistered user and the details of reply sent for each feedback.

Field Name	Field Type	Length	Field Description
IFeedListID	Int	4	List Id of feedback from user
EmailID	varchar	80	Email ID of the user who sent the feedback.
DateOfReceive	datetime	8	Date on which feedback was sent to Zendit.
Feedback	varchar	800	Feedback message.
Replied	int	4	A flag indicating whether the feedback has been replied to or not. This flag is set to "Y", if reply is sent to the feedback, else it is set to "N".
DateOfReply	datetime	8	Date on which reply was sent.
Remarks	varchar	800	The reply content.

Table Name: tblTemplate
Table Description:
This table stores the templates and its version information, which is sent to surfboard at time of logging in

Field Name	Field Type	Length	Field Description
iTemplateVer	Int	4	Template version number
vTemplate	text	16	Encrypted Template
vPassword	varchar	20	Password to decrypt the Template

Field Name	Field Type	Length	Field Description
DdateOfUpdate	datetime	8	Date of Creation of Template

Table Name: tblSendTemplate
Table Description:
This table stores the Send templates for each of the mail system for both frame and non-frame based pages.

Field Name	Field Type	Length	Field Description
lSendTempID	Int	4	Id of the send template
cMailSystem	Varchar	50	Indicating the mail system, eg: Hotmail, Yahoo.
lFrameType	Int	4	Indicates If the page is frame, non frame or iframe based
CToName	Varchar	50	HTML field name given to the "TO" field in template used by the mail system.
cCcName	Varchar	50	HTML field name given to the "Cc" field in template used by the mail system.
cBccName	Varchar	50	HTML field name given to the "Bcc" field in template used by the mail system.
cPlainText	Varchar	50	HTML field name given to the "Message" field in (plain text) template used by the mail system.
cFormatText	Varchar	50	HTML field name given to the "Message" field in (formatted text) template used by the mail system. Ex: Field name given to the

Field Name	Field Type	Length	Field Description
			Scriptlet used by the Yahoo mail system.
CFormName	Varchar	50	Name of the form containing all the information, buttons and message body
cFrameURL	Varchar	300	URL for Compose, Reply, Forward or Read frames when it's a frame based page
cMainURL	Varchar	300	URL for the Page itself
cDescription	varchar	300	A brief description about the template

Table Name: tblClickTemplate
Table Description:
This table stores the Click templates for each of the mail system for both frame and non-frame based pages. Using this we can access the different kind of submit buttons on the mailsystem pages.

Field Name	Field Type	Length	Field Description
iClickTempID	Int	4	Id of the click template
iSendTempID	Int	4	Id of the send template for which the click template is being added
iPartID	smallInt	2	The Page element for whom the template is being added
vElementType	varchar	255	Element Type

Field Name	Field Type	Length	Field Description
vAttributeName	varchar	255	Name of the Element
vAttributeValue	varchar	255	Value of that particular Attribute

Note:- The typical records for this table are listed as below

iClickTempID	=	4
iSendTempID	=	2
iPartID	=	1
vElementType	=	Input
vAttributeName	=	type
vAttributeValue	=	button

Table Name: tblReadTemplate
Table Description:
This table stores the Read templates for each of the mail system for both frame and non-frame based pages. Using this we can access the compose mail system pages.

Field Name	Field Type	Length	Field Description
iReadTempID	Int	4	Id of the read template
cMailSystem	varchar	50	Name of the mail system
iFrameType	Int	4	Type of Frame used (frame, non frame or iframe based
cHeaderStart	varchar	150	The start location in the source from which the sender information is extracted
cHeaderEnd	varchar	150	End of the header
cSenderStart	varchar	150	The start location in the source at which the sender information is embedded

Field Name	Field Type	Length	Field Description
cSenderEnd	varchar	150	End of sender tag
CTokenBegin	varchar	150	The start location in the source at which the email id of the sender is embedded
cTokenEnd	varchar	150	The tag at which token ends
cFrameURL	varchar	300	URL of the frame (if it's a frame based page)
cMainURL	varchar	300	URL of the page
cDescription	varchar	300	Description of the template

Note: tblReadTemplate is not used now for Dzending operation during signature verification. Instead it is done on the basis of Key ID.

Table Name: tblTempKeyPairPool
Table Description:
This table contains a pool of interim lock and keys generated by Zendit system. This is used for assigning temporary keys to new Email IDs. This table is populated by running a script in admin pages.
.

Field Name	Field Type	Length	Field Description
cKeyPairBlob	text	16	A blob (Binary large object) containing the key pair
CPublicKeyBlob	text	16	A blob (Binary large object) containing the public key of the key pair
cKeyID	varchar	17	A unique id of the key
iFlgUsed	int	4	A flag telling the system whether the key has been used (1 for used and 0 otherwise)

Table Name: tblTempKeyConfirmation
Table Description:
This table contains the interim keys and related info for all the interim keys which are taken out of tbltempkeypairpool and are not yet confirmed. It also has the confirmation IDs for the users who have done quick registration.

.

Field Name	Field Type	Length	Field Description
cEmailID	varchar	250	Email id associated with the temporary key
cConfirmationID	varchar	20	Confirmation ID sent to the target user
cKeyID	varchar	17	A unique Key ID
iMailSent	int	4	A flag depicting the success of Confirmation mail sent to the user. (0 if fail and 1 otherwise)
cSenderEmailID	varchar	2500	Email id of the sender who requested the temp key
cLoginID	varchar	20	Login id of the sender
dtCreate	datetime	8	Date of creation of temp key

Table Name: tblConfirmedTempKeys
Table Description:
This table contains the Confirmed interim keys and related info

.

Field Name	Field Type	Length	Field Description
cLoginID	varchar	20	Login ID of the user
cEmailID	varchar	250	Email id associated with the temporary key

Field Name	Field Type	Length	Field Description
cKeyPairBlob	text	16	The Key pair blob (Private key + public key)
cKeyID	varchar	17	A unique Key ID

Table Name: tblDPL
Table Description:
This table contains the names and info of persons on US federal government's denied person's list.
.

Field Name	Field Type	Length	Field Description
iListID	int	4	Login ID of the user
cName	varchar	250	Full name of the person
iFlag	smallint	2	Flag indicating whether it's a person or a company's name (1 for person and 2 for company)
cAddress1	varchar	250	Address of the person/company
cAddress2	varchar	250	Address of the person/company
cAddress3	varchar	250	Address of the person/company
cCity	varchar	250	City of person/company
cState	varchar	250	State of person/company
cCountry	varchar	250	Country of person/company (required)
cZip	varchar	50	Zip code of person/company
cEffectiveDate	varchar	250	Restrictions applicable from

Field Name	Field Type	Length	Field Description
			date(required)
cExpirationDate	varchar	250	The date when restrictions expire(required)
cFRC	varchar	250	The FRC issued
CFRC_Date	varchar	250	Date of issuance of FRC

Table Name: tblListLoginIDs
Table Description:
This table contains the Login Ids and corresponding unique List ids of registered users
.

Field Name	Field Type	Length	Field Description
iListID	int	4	A unique id generated for each user
cLoginID	varchar	20	Login id of the user
dtLastAccessed	datetime	8	Date when user last accessed the system

Table Name: tblListEmailIDs
Table Description:
This table contains the EmailIDs and corresponding Keys of registered users
.

Field Name	Field Type	Length	Field Description
iListID	int	4	A unique id generated for each Email address
cEmailID	varchar	250	Email id
cName	varchar	250	Name of the user associated

Field Name	Field Type	Length	Field Description
			with email id
cKeyID	varchar	17	ID of the key associated with the email id

Table Name: tblLoginEmailIDs
Table Description:
This table contains the Email IDs of registered users.

Field Name	Field Type	Length	Field Description
cLoginID	varchar	20	Login Id of the user
cEmailID	varchar	250	Email ID associated with the user
iFlag	int	4	A flag determining whether the email address has an associated public lock or zend generated lock and key

Table Name: tblMailContent
Table Description:
This table contains the mail contents that are sent to the user for different scenarios.

Field Name	Field Type	Length	Field Description
iContentID	int	4	A unique id for the mail content
cMailFor	varchar	250	Purpose of the mail
cSubject	varchar	250	Content of mail subject
cMailContent	varchar	5000	Content of mail body
cMailDesc	varchar	500	Brief Description about the

Field Name	Field Type	Length	Field Description
			mail

Table Name: tblMailsZent
Table Description:
This table keeps track of the number of mails zent to each mail system everyday
.

Field Name	Field Type	Length	Field Description
cMailSystem	varchar	50	Name of the mail system
dtMailZent	datetime	8	Date of zending mails
iMailsZent	int	4	Number of mails zent on that particular day to that particular system

Table Name: tblMailsZentLogin
Table Description:
This table keeps track of the number of mails zent by each user.

Field Name	Field Type	Length	Field Description
cLoginID	varchar	20	Login id of the user
iMailsZent	int	4	Number of mails zent by the user

Table Name: tblServerMap
Table Description:
This table Maps Servers names to unique Hash IDs.

Field Name	Field Type	Length	Field Description
iHashID	int	4	Hash id to be mapped to

Field Name	Field Type	Length	Field Description
cServerName	varchar	100	SQL Server Name

Table Name: tblSurfBoardDownload
Table Description:
This table shows number of surfboard downloads for a particular day

Field Name	Field Type	Length	Field Description
iDownloadCount	int	4	Number of downloads for a particular day
DateOfDownload	datetime	8	Date of download recording

Table Name: tblLoginStatus
Table Description:
This table tracks the user's login status once he is logged in

Field Name	Field Type	Length	Field Description
cLoginID	Varchar	20	Login Id of the user
cStatusHash	Varchar	50	A hash of session id and time stamp
iSourceFlag	int	4	A flag denoting whether the user logged from web or surfboard
dtLastAccessed	Datetime	8	The timestamp of the last user interaction with the server

Table Name: tblReqPwdConf
Table Description:
This table contains the Confirmation IDs for those who have requested for password as they had forgotten it.

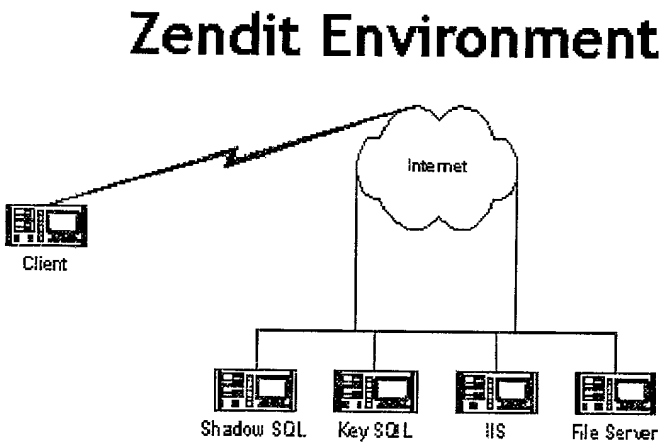
Field Name	Field Type	Length	Field Description
cLoginID	Varchar	20	Login Id of the user
cEmailID	Varchar	250	Primary Email Address of the user
cConfID	Varchar	20	Confirmation ID
iMailSent	Int	4	The flag that indicates whether the mail is sent or not.
dtDate	Datetime	8	Date on which the request is made

Table Name: tblHelpMapping
Table Description:
This table contains the mapping between the Mapping ID and the corresponding help page.

Field Name	Field Type	Length	Field Description
cID	Varchar	15	Mapping ID
cPageName	Varchar	100	The corresponding help page.

3. Zendit System Design

The diagram depicts the Zendit architecture as proposed.



3.1 Hashing implementation

The hashing algorithm is applied to ensure that data is spread evenly across different SQL Server machines. The maximum number of machines is assumed to be 100. A mapping between a hash ID and SQL Server name is maintained.

When a user registers with Zendit, the login ID is passed as a parameter to the hashing function. After hashing, a hash ID is generated which would be a number between 1 and 100. Depending on the hash ID generated, the user data is moved to the SQL Server that is mapped with that ID.

From there on, hashing is performed on the login IDs to identify the SQL Server on which the user data is available.

The hashing would be performed on email IDs also.

3.2 Confirmation process

Confirmation process could be classified as

1. Registration confirmation
2. Certificate confirmation

3.2.1 Registration confirmation

When the user registers successfully, a mail is sent to the primary email ID of the user. The mail would contain a confirmation ID that the user is required to submit at the website.

A record is inserted into the tblRegistrationConfirmation table with the login ID, the confirmation ID and date of registration. The user record in the tblRegisteredUsers is marked as non-confirmed.

When the user submits the confirmation ID, the tblRegistrationConfirmation is queried for a record with matching login ID and confirmation ID. If found, the record in the tblRegisteredUsers table is marked as confirmed and the corresponding record in the tblRegistrationConfirmation table is deleted.

3.2.2 Certificate confirmation

When a newly created certificate is sent to the Zendit server along with an email ID, the following scenarios are handled:

Key for an email ID does not exist

A record is inserted into the tblNonConfirmedCerts table with the login ID, confirmation ID, key ID and other relevant data.

A mail will be sent to the email ID associated with the key ID. The mail would contain a confirmation ID. The record is marked as 'mail sent' in the tblNonConfirmedCerts table once the mail is sent.

A record is inserted into the tblKeyConfirmation table with the login ID and key ID values along with other relevant information.

Once the user submits the confirmation ID at the Zendit server (this process is done through the web site), the login ID and key ID combination is searched for in the tblKeyConfirmation table. If found, record from the tblNonConfirmedCerts with this combination is moved to (Inserted) the tblConfirmedCerts table and is deleted from the tblNonConfirmedCerts. The record in the tblKeyConfirmation is also deleted.

Key for an email ID that exists in the Zendit Server (tblConfirmedCerts)

If the key is for the same Login ID, a message would be displayed to the user informing about he key for the email ID already existing in the Zendit Server, with an option to replace it.
If the user chooses to replace, the new key-email ID information is inserted into the tblConfirmedCerts table and the older certificate will be moved to tblBadCerts.

If the key is associated with different Login ID, a message would be displayed that the Email address is associated with another login ID and cannot be published.

3.3 Security

3.3.1 Security on Pages

The pages that process SurfBoard requests would check to see if the referrer URL is 'NULL' or the domain of the referrer URL would be the Zendit domain. As soon as the user logs in the Zendit system, the session id will be hashed and stored in tblStatusHash. This statushash would be passed as parameters to all the interface pages that will validate with the database.

The pages on the web site that act as an interface for user activity in the site, the identity of the user logged in would be maintained in a session based cookie. For all requests that require the user to be logged in, the pages would check the cookie to find out if the user has logged in. These pages also would look for the referrer URL to check if the requests are being made from the Zendit domain itself. Also when the user logs in the web vault, the session id will be hashed and stored in tblStatusHash. This status hash will be stored in a cookie and the value of this cookie will be validated with that in the database.

3.3.2 Database Security

The user password is to be hashed using a hashing algorithm and stored in the database. This is to prevent unauthorized people accessing the password from the database. This prevents any security hazard for the users, as any person having access to the password in the database will be able to retrieve any personal information and more importantly execute any privileged task such as deleting the user's backed up key pairs or invalidating the credentials of the user by changing the user's primary email address. Thus the user password is hashed and the password hash is stored in the database.

Note: For further details, please refer to the User Authentication and Login Status document attached under the section 3.4

3.4 User authentication and login status

The way the user is authenticated and method with which the user login status is maintained at the server is explained in the attached document. This is to ensure that no other person impersonates the user. The design of this process and the issues involved are discussed in the attached document.

"User Authentication
And Login Status.doc"

3.5 Denied Persons List

The US federal government restricts the export of security related software like Zendit to certain countries. Also, a list of persons that is maintained by the federal government are to be denied the services of the Zendit system. The implementation details and the issues involved are discussed in the attached document.

"DPL High Level Design.doc"

3.6 Zend generated lock and key

When the user tries to send an encrypted message to an email address, which does not have an associated public lock in the Zendit directory of certificates at server database, the server generates an lock and key for the email address and uses the public lock to encrypt the message. The encrypted message is sent along with another mail that asks the receiver to register with the Zendit system and to download the Zend generated lock and key. During the lock and key download process, the user is asked to change the password associated with the lock and key.

3.7 COM object under MTS for transactions

All database transactions are implemented in an ActiveX DLL, which is placed under MTS. This provides a stable and reliable way of implementing large-scale transactions as well as complying with the three-tier architecture. The design, implementation and the security issues involved are detailed in the attached document.

"COM Design Document.doc"

4. Zendit Web Site

The purpose of this section is to provide an overview of the high-level design implementation of the pages in the Zendit Website. This section details the behavior of each of the pages in the site and the backend processing that occurs for the user actions on the site.

The Zendit Website is divided into three sections:

4.1 User Pages.
4.2 Administration Pages.
4.3 Interface pages to SurfBoard.

4.1 User Pages

The Zendit website provides an interface for users to register and download the surfboard. Registered users can view their personal 'Vault' where their related information is displayed. The different options available to the user are:

1. Registration
2. Key confirmation

3. Login
4. Search
5. Feedback
6. View Vault
   a. Personal Information
   b. Personal settings
   c. Files uploaded(not applicable now)
   d. Key Pairs uploaded
   e. View Certificates
   f. Change password
   g. Download surfboard.
6. Forgot password

The following sections describe each of the options available to the user in detail.

4.1.1 Registration

Description

There are four different ways with which the user can register with the Zendit system. The method. The regular registration process is discussed below. The other methods are discussed after the basic registration process.

The page collects user's information for registration. The user is also provided with an option to confirm the registration.

Refer Functional Specification document for details regarding the registration process.

Validations

During registration:
- Check for mandatory fields.
- Check for the data type.
- Check for minimum and maximum characters/value required.
- Check for the format of the primary email ID.

During registration confirmation:
- Check if confirmation ID is entered.

Database activities

The following table is used during the registration process.

1. tblRegisteredUsers

Actions performed:
- Check for login ID.
- If login ID exists, a message is displayed to the user to choose a different login ID.
- If login ID does not exist, a new record is inserted into the table. The password is hashed and this password hash is stored in the table.
- A mail is sent to the user with a confirmation ID. The email ID to which the mail is sent will be the primary email ID of the user.

The mail that is sent to the user would contain the URL for confirmation with the login ID and the confirmation ID as parameters. This facilitates the user to be taken to the confirmation page and get the registration confirmed.
For e.g., the URL would look like

http://www.Zendit.com/confirm.asp?loginID=john&confID=x2we4fdvc4
The user needs to submit his password during his registration confirmation.

The following tables are used during the registration confirmation process.

1. tblRegistrationConfirmation
2. tblRegisteredUsers

Actions performed:
- Check for the login ID and confirmation ID combination in the tblRegistrationConfirmation table.
- If combination does not exist, display message to the user.
- If combination exists, mark the related record for that login ID in the tblRegisteredUsers table as confirmed and delete the record from the tblRegistrationConfirmation table.

The other registration methods are as follows.

Quick Registration

The Zendit system generates a lock and key for the user when user registers with the system. The user can download this lock and key after confirming the registration through the ZPanel. The only restriction is that the primary email address provided in the registration should not be associated with another user in the Zendit server.

Vaccine Registration

The Zendit system generates a lock and key for those email address for which no lock is associated in the Zendit server while someone is trying to send an encrypted mail to the email address using the Zendit system. Along with the encrypted mail, another mail is sent from the Zendit system asking the receiver to confirm that the email address belongs to the receiver. When the receiver is not a registered user of the Zendit system and tries to confirm the email address, then the receiver has the option of registering with the system. There is no registration confirmation process involved with this registration as the user has already confirmed that the email address is associated with the registered login.

Surfboard Registration

At the end of the Zendit client installation, the setup provides the user with the option of registering with the Zendit system. When the user chooses to register, the quick registration process is initiated.

4.1.2 Certificate Confirmation

Description

The page provides an option for the user to confirm the key created.

Refer the FS document for details regarding the key creation process.

The URL for confirming the Key creation would contain the email ID and Key ID as a parameter. For e.g.,
http://www.zend.com/confirm.asp?loginID=john&KeyID=12345678

Validations

- Check if the confirmation ID is entered.

Database activities

The following tables are used during the key confirmation process.

1. tblKeyConfirmation
2. tblConfirmedCerts
3. tblNonConfirmedCerts

Actions performed:

- Perform the hashing on the login ID to identify the SQL Server to be accessed.
  - Check for the key ID and confirmation ID combination.
  - If combination does not exist, display message to the user.
  - If combination exists, insert a record into the tblConfirmedCerts table with the values from the tblNonConfirmedCerts for that key ID.
  - Delete the record from the tblNonConfirmedCerts table and tblKeyConfirmation table for that key ID.

4.1.3 Login

Description

This page provides an option for the user to login to the site to view the web vault.

Validations

- Check if the login ID and password are entered.
- Check for the minimum and maximum characters required.

Database activities

The following table is used during the login process.

1. tblRegisteredUsers

Actions performed:

- Hash the password and get the password hash
- Check for the login ID and password hash combination.
- If combination does not exist, display message to the user.
- If combination exists, hash the session ID, date and time and get the status hash. Drop a session-based cookie onto the user machine with the status hash value in it.
- This cookie would be read on all the pages to check if the user has logged in.
- Display the personal vault page.

4.1.4 Search

Description

This page facilitates the user to search for a particular certificate based on name, email ID or the key ID.

Actions performed

For Name search:
The ListEmailIDs table in the admin server is searched for an exact match of the name provided

For email ID search:
The ListEmailIDstable in the admin server is searched for an exact match of the email ID provided.

For key ID search:
The tblListEmailIDs table in the admin server is searched for an exact match of the key ID provided.

4.1.5 Feedback

Description

This page facilitates any user to send his feedback about the Zendit system. The user need not necessarily be logged in to the system.

Actions performed

The email ID and the feedback from the user are inserted in the tblFeedback table along with the date.

4.1.6 Vault

Description

This page provides an option for the user to view all the information regarding the user that is available at the Zendit Server.

Database activities

The following table is used while accessing the vault.

1. tblRegisteredUsers

Actions performed:
- Access the cookie to see if the user has logged in.
- If not logged in, display message to the user.
- If logged in, display the page with appropriate links.
- Display the remaining free space available to the user and total mails zent by the user.

These values are obtained from the tblRegisteredUsers table.

4.1.7 Personal information

Description

This page is accessed from within the Vault page. This page provides an option for the user to view the personal information that is entered during registration. The user is also provided with an option to update this information.

Validations

- Check for mandatory fields.
- Check for the data type.
- Check for minimum and maximum characters/values required.
- Check for the format of the primary email ID.

Database activities

The following table is used while accessing the vault.

1. tblRegisteredUsers

Actions performed:
- Access the cookie to see if the user has logged in.
- If not logged in, display message to the user.
- If logged in, display the page with values shown in the appropriate fields.
- On submit of this page, update the information back into the record for that login ID.

Case handled
When the user changes the primary email ID, a process similar to the registration is carried out. The user is sent mail both to the old primary email ID and the new email ID. While the mail sent to the new email ID contains the confirmation ID, the old email ID is sent a mail informing the user that a request for change of primary email ID was made. The user would not be able to login into either the SurfBoard or the vault till the confirmation ID is submitted.

4.1.8 Personal settings

Description
This page is accessed from within the Vault page. This page provides an option for the user to choose the personal information fields that could be made available to other Zend users.

Database activities

The following table is used while accessing the vault.

1. tblRegisteredUsers
2. tblUserProfile

Actions performed:

- Access the cookie to see if the user has logged in.
- If not logged in, display message to the user.
- If logged in, display the page with values shown in the appropriate fields. Display checkboxes next to each field.
- On submit of this page, check for the fields that are selected. Store this information in tblUserProfile table.

The selected fields are stored as a string delimited by commas in a single field.

For e.g., if the user selects First Name, Phone and Address, this information is stored as FName,Phone,Address in a field in the record for that login ID.

4.1.9 Uploaded files

Note: This option is not present now.

Description

This page is accessed from within the Web Vault page. This page provides an option for the user to view the file information backedup to the Zendit server. The user is also provided with an option to download or delete the uploaded files.

Database activities

The following table is used while accessing the vault.

1. tblFileUpload

Actions performed:

- Access the cookie to see if the user has logged in.
- If not logged in, display message to the user.
- If logged in, select all the records from the tblFileUpload table for that login ID and display the information on the page. The file names would appear as hyperlinks.
- Clicking on a file name link would download the file to the client machine.

Note: This feature is not implemented in the current feature.

4.1.10 Backed Up Locks and Keys

Description

This page is accessed from within the Web Vault. This page provides an option for the user to view the locks and keys that are backed up from the SurfBoard onto the Zendit server. The user is provided with an option to delete a lock and key by selecting the lock and key to be deleted and clicking the delete button.

Database activities

The following table is used while accessing the vault.

1. tblKeyUpload

Actions performed:
- Access the cookie to see if the user has logged in.
- If not logged in, display message to the user.
- If logged in select all the records from the tblKeyUpload table for that login ID and display the information on the page.

When the user selects a lock and key to delete and clicks the delete button, this information is posted to another page, which would accept the login ID and key ID as parameters and run a stored procedure to delete the record from the tblKeyUpload table.

Note: The user is not provided with an option to restore the locks and keys from the site. The locks and keys can be restored vault.

4.1.11 View Certificates

Description

This page is accessed from within the Web Vault. This page provides an option for the user to view the certificates that are associated with the user on the Zendit server. The user can also delete the published lock from the web vault.

Database activities

The following tables are used while accessing the vault.

1. tblRegisteredUsers
2. tblConfirmedCerts

Actions performed:
- Access the cookie to see if the user has logged in.
- If not logged in, display message to the user.
- If logged in, obtain information regarding the email IDs associated with the login ID from the tblLoginEmailID table.
- Perform the hashing on the email IDs to identify the SQL Server to be accessed.
- Select the certificate-related information for the email IDs from tblConfirmedCerts table in the appropriate SQL Servers and display this information on the page. The EmailIDs and key IDs would appear as hyperlinks. Clicking on those would display the certificate information in a simple format.

4.1.12 Change password

Description

This page is accessed from within the Web Vault. This page provides an option for the user to change the password for login.

Validations

- Check for old password value.
- Check for new password value.
- Check for confirmed password value.
- Check for minimum and maximum characters required.
- Check if the new password value and the confirm password value match.

Database activities

The following table is used while accessing the vault.

1. tblRegisteredUsers

Actions performed:
- Access the cookie to see if the user has logged in.
- If not logged in, display message to the user.
- If logged in, check if the old password matches with the entered old password value.
- If the values don't match, display message to the user.
- If values match, update the password value with the new password value after hashing it.

4.1.13 Forgot password

Description

The user would not be able to login into the SurfBoard or the vault in the web site if the password is forgotten or lost. The user can request for the password under such circumstances. The user is required to submit the login ID and his primary Email ID. If the combination is valid, a confirmation ID is generated and inserted into tblReqPwdConf along with the login ID and a mail is sent to the user with a link.
On clicking of the link, the confirmation ID is validated and the user needs to submit his new password. The password will be hashed and updated in the database.

Database activities

The following table is used while accessing the vault.

1. tblRegisteredUsers
2. tblReqPwdConf

Actions performed:
- Check for the login ID and primary Email ID combinationin tblRegisteredUsers table.
- Generate Confirmation ID and insert into tblReqPwdConf.
- On click of the link, check for the validity of login ID, confirmation ID in tblReqPwdConf.
- If valid, the new password is got from the user, hashed and updated.

4.2 Administration Pages

The administration pages are a set of web pages used to manage admin related activities such as database related settings, getting statistical information and searching for user information..

Access Mechanism will be **Windows NT challenge-response authentication.**

4.2.1 Menu Page

Description

The page contains the various menu options that are available to the admin. The authorized users using Windows NT challenge-response authentication can access this page. The various menu options are:

- Search
- Statistics
- Feedback
- Last Accessed
- Admin Settings
- Templates
- Scripts
- Mailers
- Restrictions
- Help
-

4.2.2 Search / User Information Page

Description

This page facilitates the admin to search for a particular user based on the login ID, email ID or the Key ID.

Actions performed
For login ID search:
The search is done on a **pattern** match basis and returns all users' login IDs containing the search string provided. The tblListLoginIDs table in the admin server is searched to obtain the result.
For email ID search:
The search is done on a **pattern** match basis and returns all the Email IDs containing the search string provided.The tblListEmailIDs table in the admin server is searched to obtain the result.

For key ID search:
The tbltblListEmailIDs table in admin server is searched for an exact match of the key ID provided.

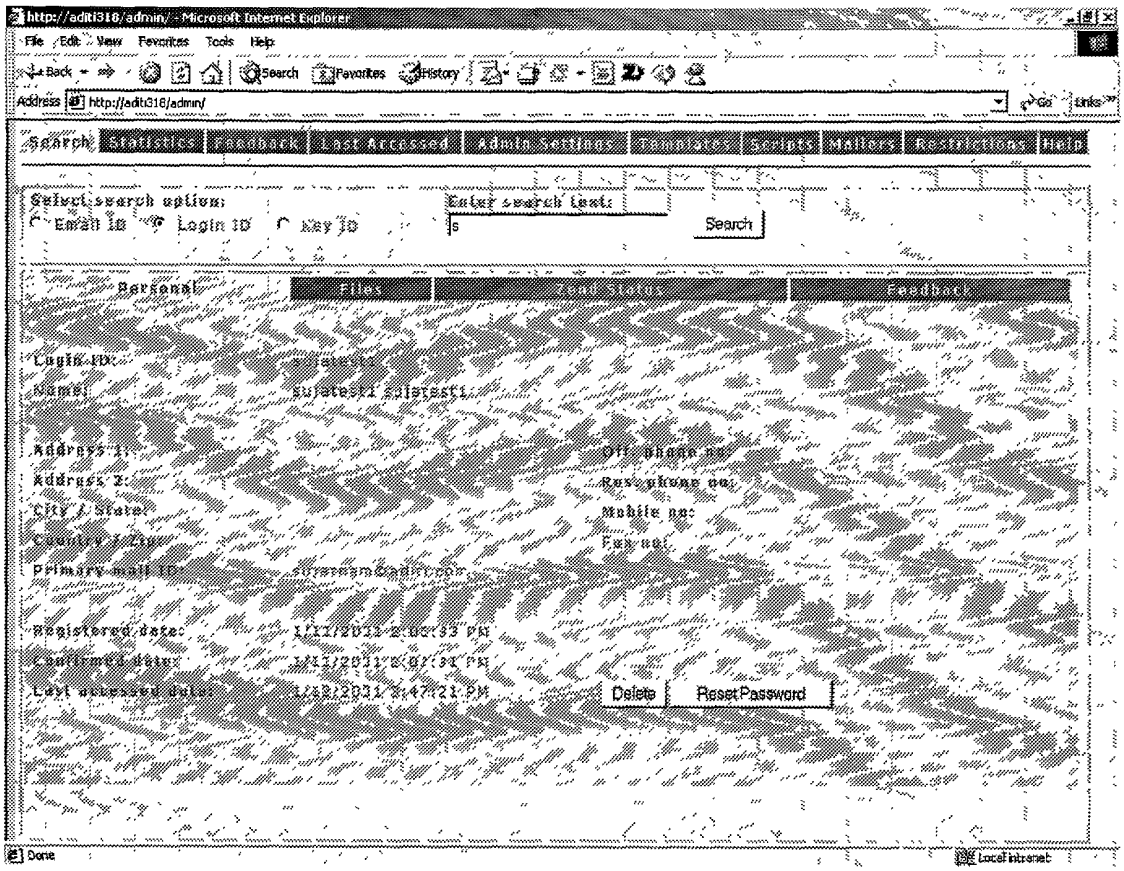
The user information is divided under the tabs as follows:

Personal

The personal details of the user such as the name, address, phone numbers, registration date, confirmation date and the last accessed date are displayed under this tab. The admin will have provisions for resetting user password as well as deleting the user account.

The following table is accessed for personal information.

1. tblRegisteredUsers

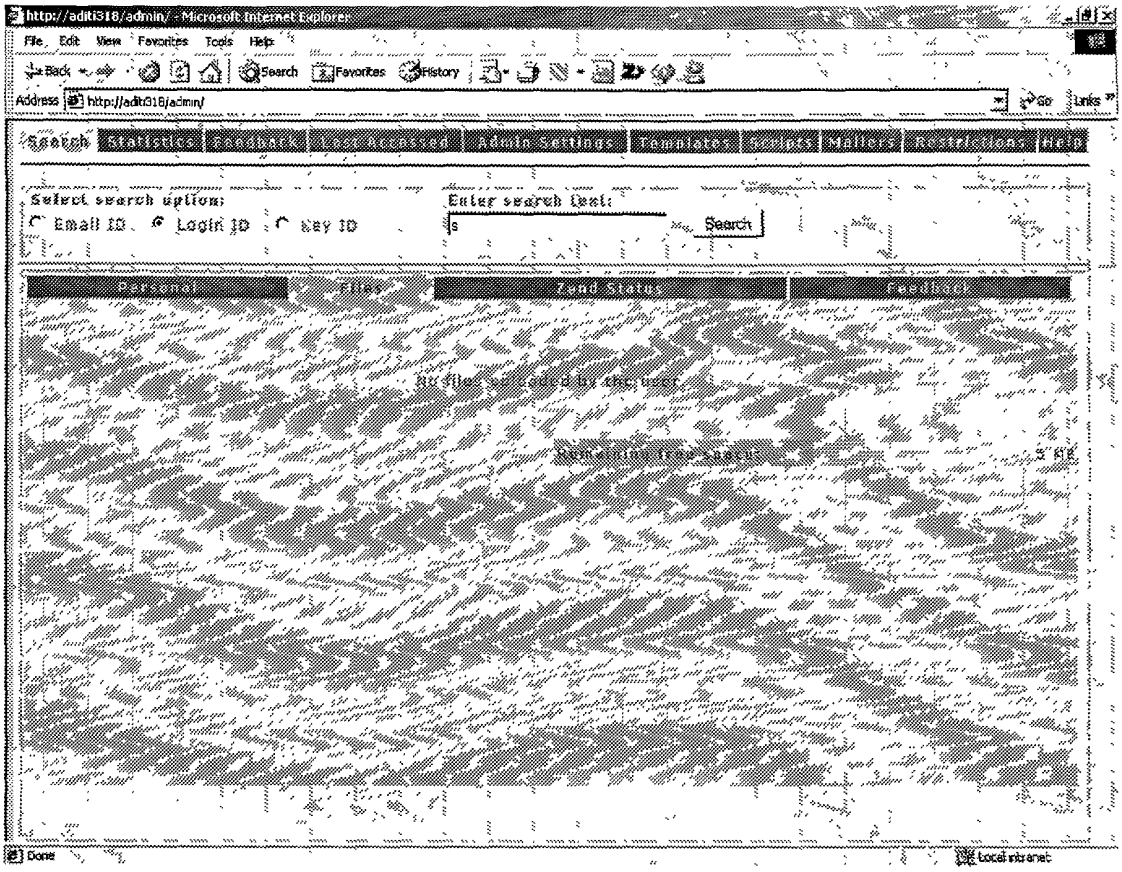


Files

The uploaded file details such as the name, size, uploaded date, total used space and free space left are displayed under this tab.

The following table is accessed for uploaded file information.

1. tblFileUpload

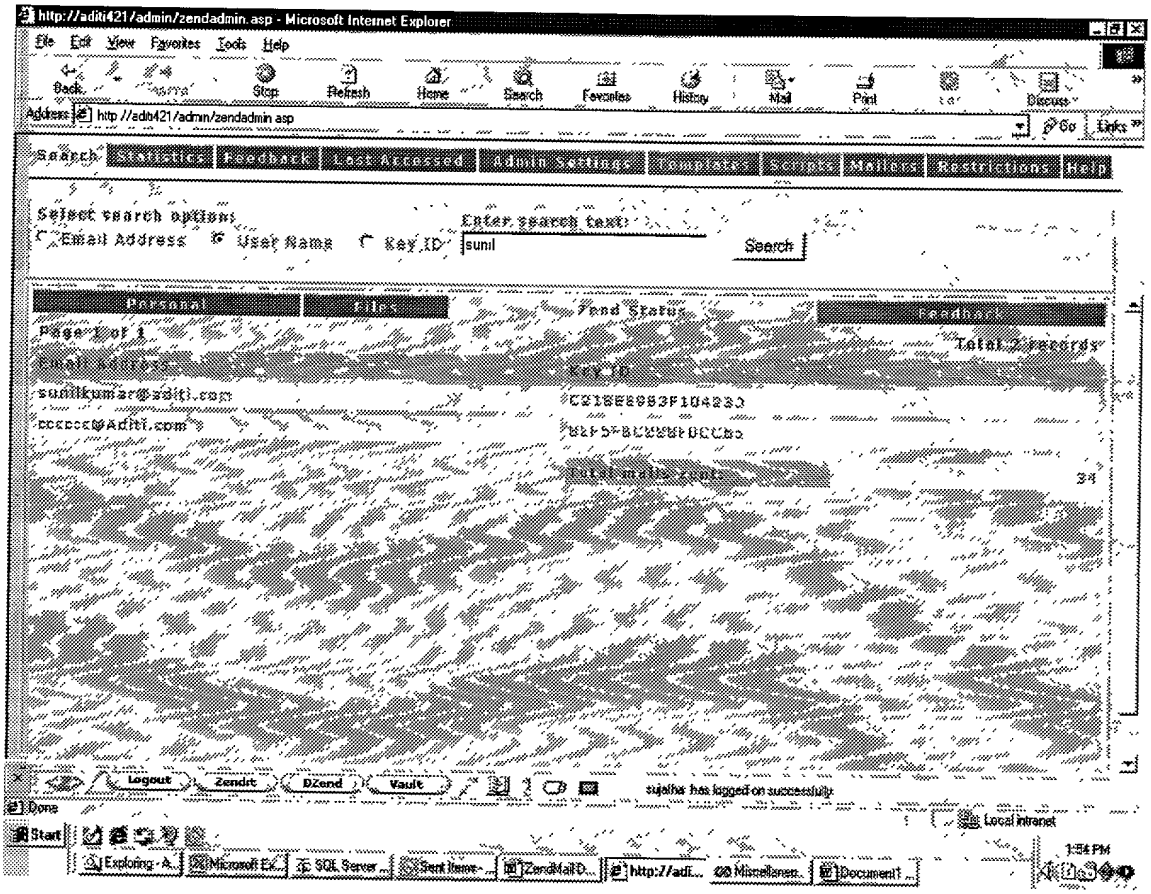


Zend Status

The user's associated email IDs, corresponding key IDs and the number of mails zent by the user are displayed under this tab.

The following table is accessed for key and mails zent information.

1. tblConfirmedCerts

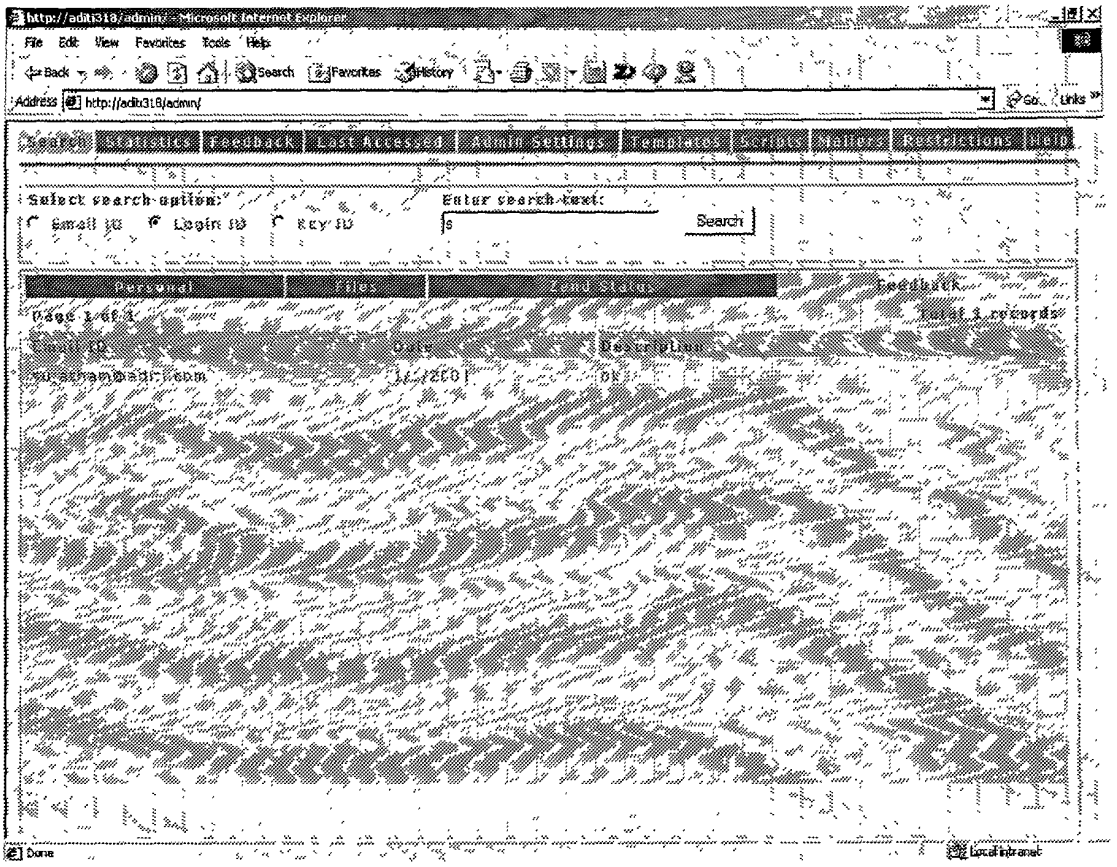


Feedback

The details of all feedbacks sent by the user are displayed under this tab.

The following table is accessed for feedback information.

1. tblFeedback



4.2.3 Statistics Page

Description

This page facilitates the admin to view various statistics of the Zendit system. The admin can select the queries and the date ranges for which he needs to see the results. The admin server will be accessed in sequence for all statistical information that the queries generate. The details of the statistics that the admin can view are as follows:

- Zent mails via a particular mail system

This query allows the admin to view the number of mails that had been sent using SurfBoard through a particular mail system (ex – Hotmail, Yahoo). The admin selects the mail system and provides the date range for which the statistics has to be shown. The tblMailsZent table is accessed to fetch the result.

- Number of confirmed registrations

This query accesses the tblRegisteredUsers table to find out the number of people who have registered and have confirmed by checking the confirmed flag field for a given date range.

- Total number of Published Locks

This query accesses the tblConfirmedCerts table to fetch the total number of digital IDs that have been registered with the Zendit system.

- Average email IDs per user

This query accesses the tblConfirmedCerts table to find out the average number of digital IDs per person that has been registered with the Zendit system.

- Average number of files per user

This query accesses the tblFileUpload table to fetch the average number of files uploaded per user.

- Average file space used per user

This query accesses the tblFileUpload table to fetch the average uploaded file space used per user.

- Total Number of Surfboard downloads

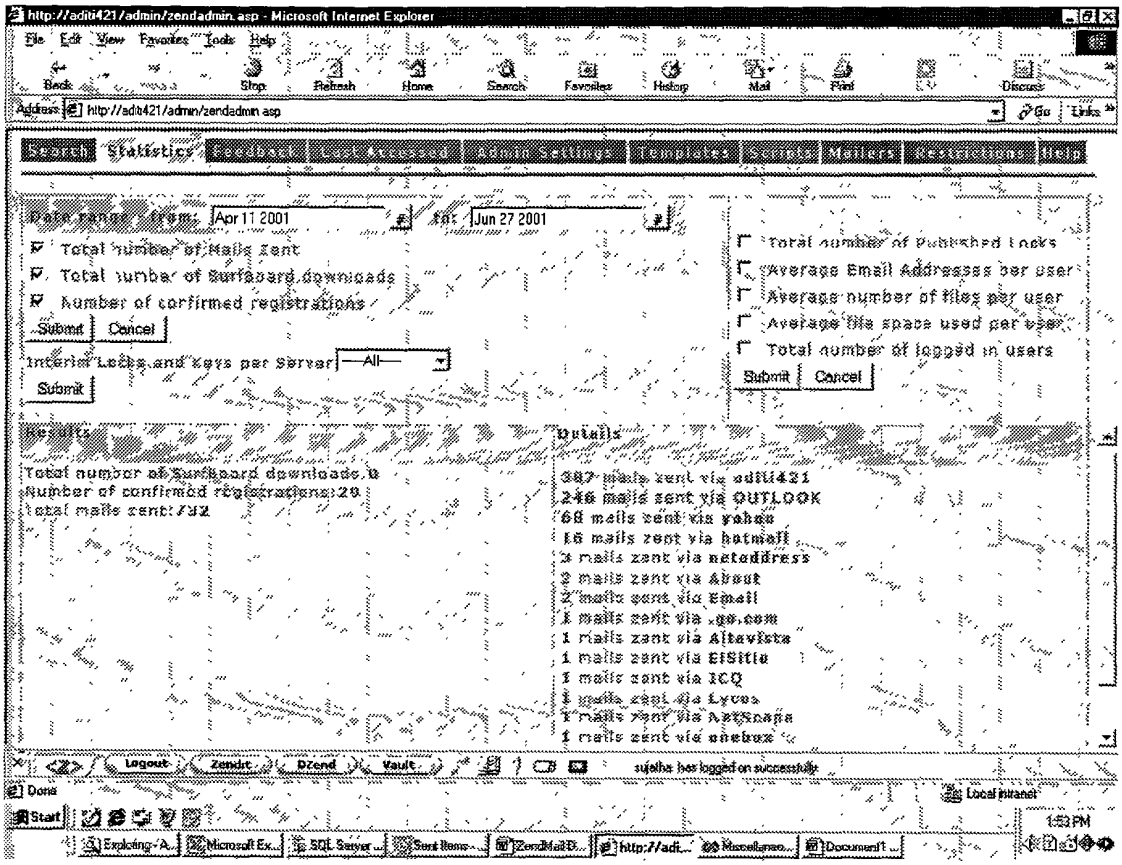
This query accesses the tblSurfboardDownload to find the number of surfboard downloads that have happened during a given date range.

- Total number of logged in users

This query accesses the tblLoginStatus table to find the number of users who have logged into website and through surfboard at any point of time.

- Number of Interim Lock and Keys in the server

This query accesses the tblTempKeyPairPool in the selected server to find the number of unused interim locks and keys.



4.2.4 Feedback Page

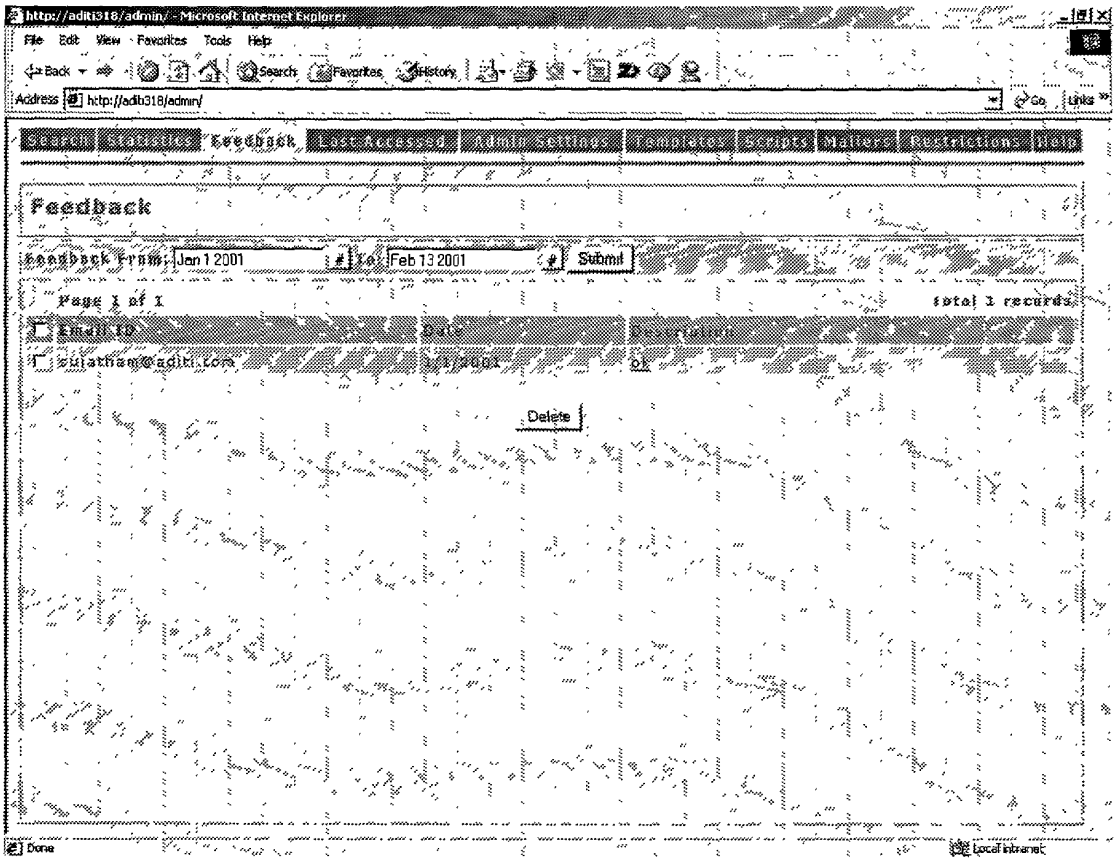
Description

This page allows the admin to view all feedback that was received by the Zendit system on a given date range. When the admin clicks on a particular feedback, a new window that pops up will provide the interface for the admin to do the following:

- To delete the feedback
- To mail back with the reply to the user.

The following table is accessed for feedback information.

1. tblFeedback



4.2.5 Last Accessed Page

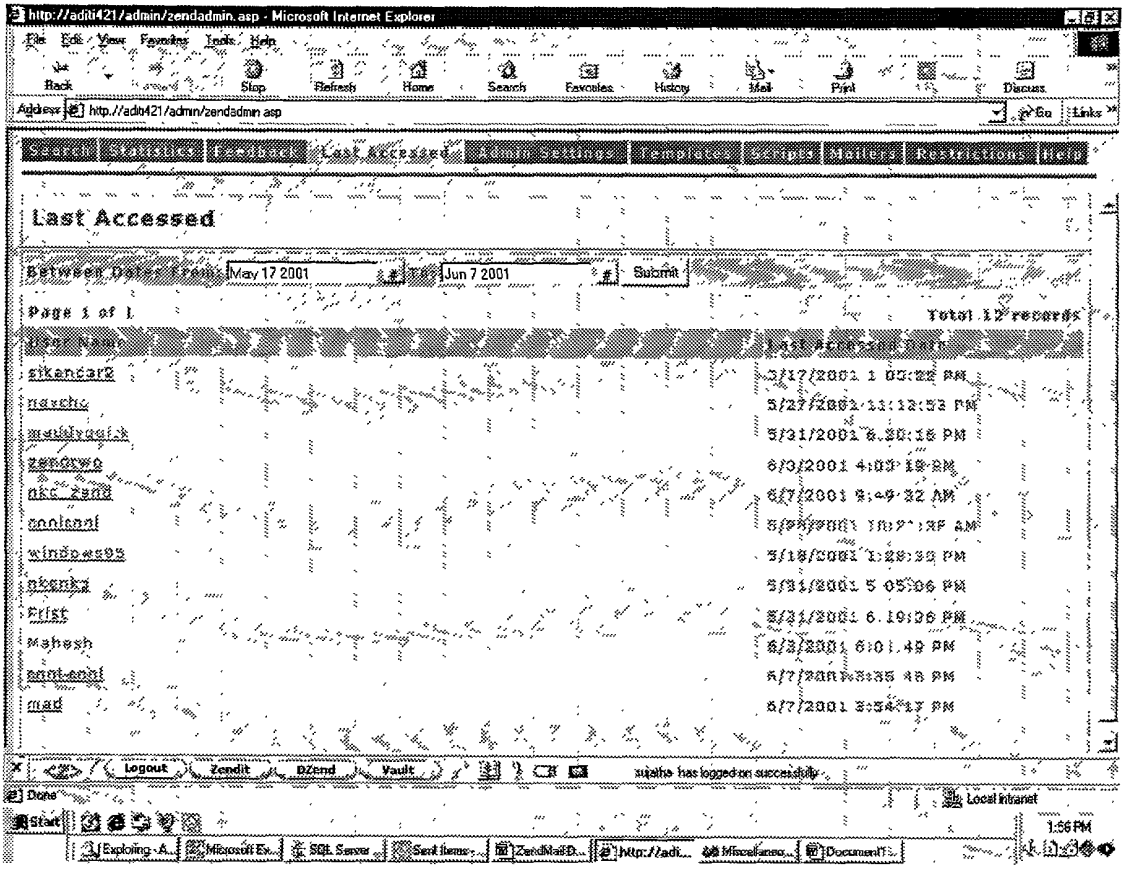
Description

This page allows the admin to view all users who last accessed the Zendit system between two dates. The admin is provided with the interface to do the following activities:

- Reset the password and send a mail.
- Delete the user accounts.

The following table is accessed for last accessed information.

1. tblListLoginIDs



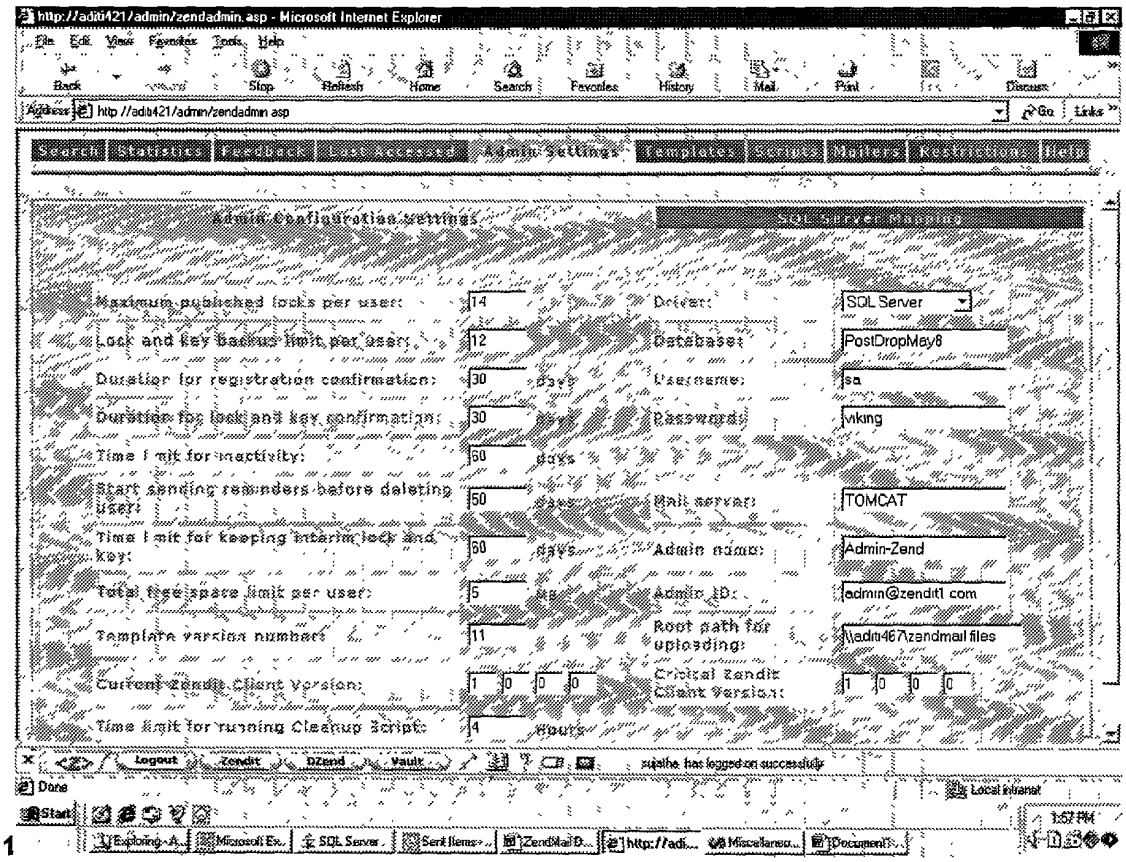
4.2.6 Admin Settings Page

Description

This page allows the admin to change settings related to the Zendit system. The settings that can be changed are as follows:

- The total file space limit per user.
- The root path for file uploads.
- Locks and Keys per user.
- Maximujm published locks per user.
- Duration for key pair confirmation.
- Duration for registration confirmation.
- Duration for keeping the intrim lock and key in the server
- Time limit for inactivity.
- Time limit for sending reminder mails to the user regarding last accessed date.
- The template version number.
-
- DSN string.
- Current Zendit Client Version.
- Critical Zendit Client Version.
- Time limit for running cleanup script (to cleanup the records in tblLoginStatus who have logged in but have not used the system more than the specified time.)

The following table is updated for admin settings.
TblAdminSettings



1

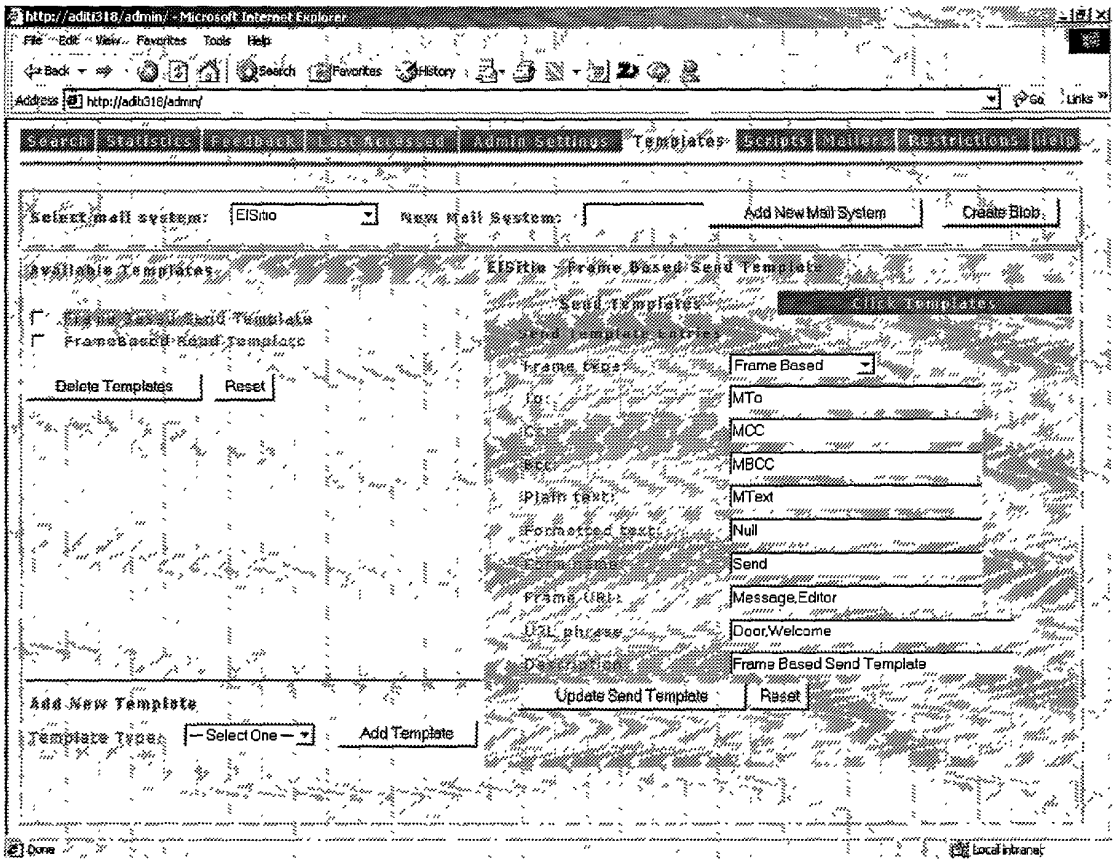
4.2.7 Templates Page

Description

This page allows the user to update the templates for mailsystems, Frame based and Non Frame based usage. The admin will also be able to add new mail system templates. After updating the templates, the template blob is created and when the application variables are refreshed, the template version number in tblAdminSettings is increased by one.

The following table is accessed for template information.

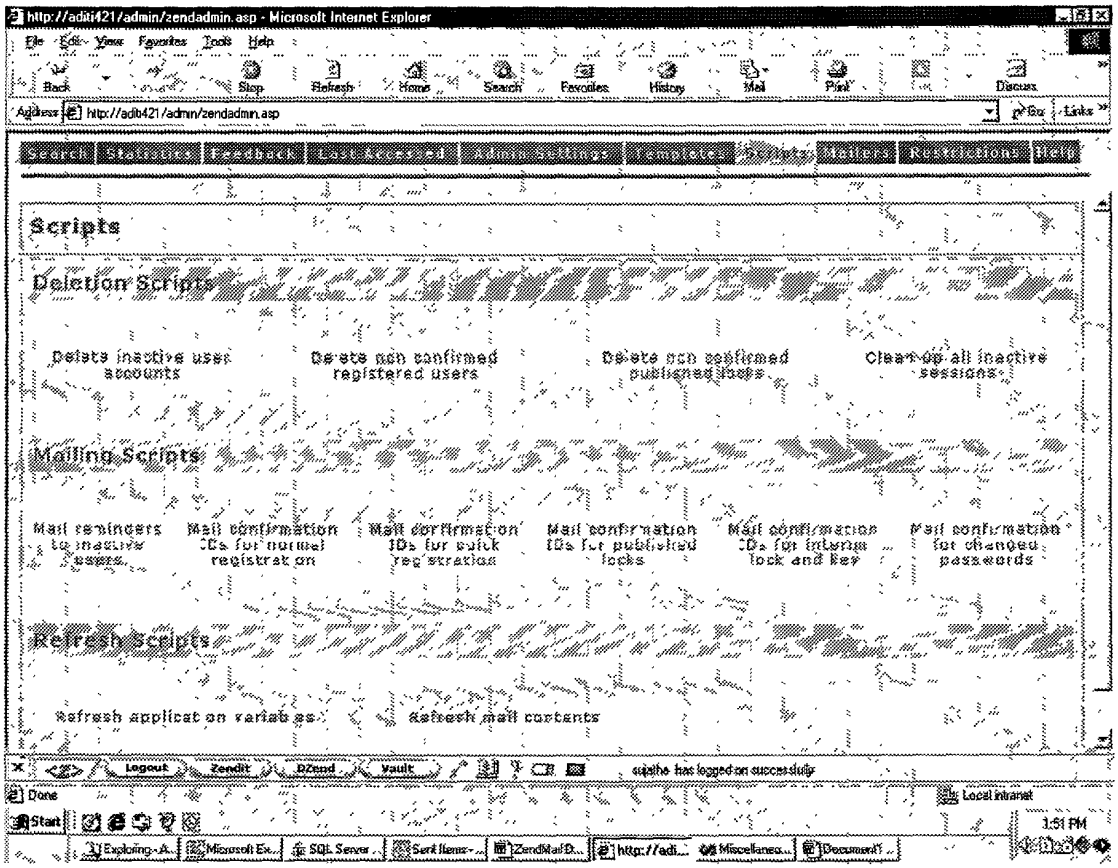
1. tblTemplate
2. tblSendTemplate
3. tblReadTemplate
4. tblClickTemplate



4.2.8 Scripts

Description

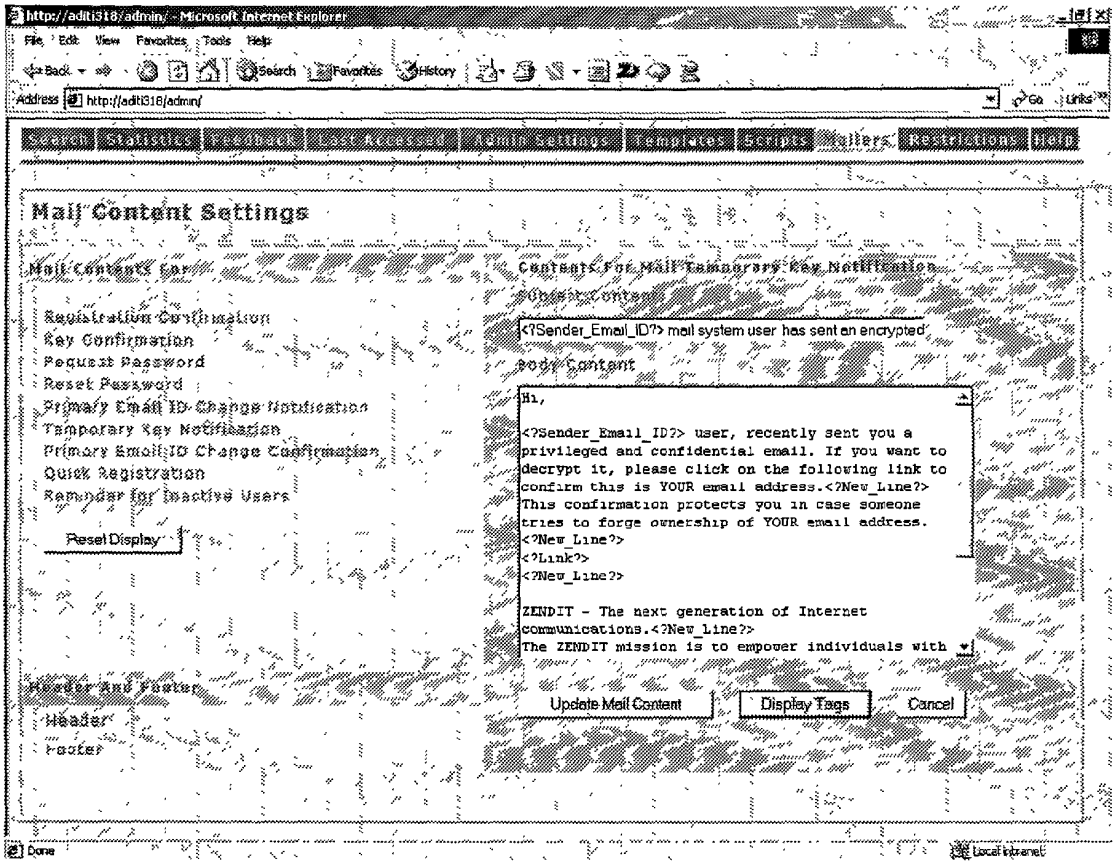
This page contains the manual scripts needed for performing various maintenance and cleanup related tasks and also the scripts to send mails that were not sent .



4.2.9 Mailers

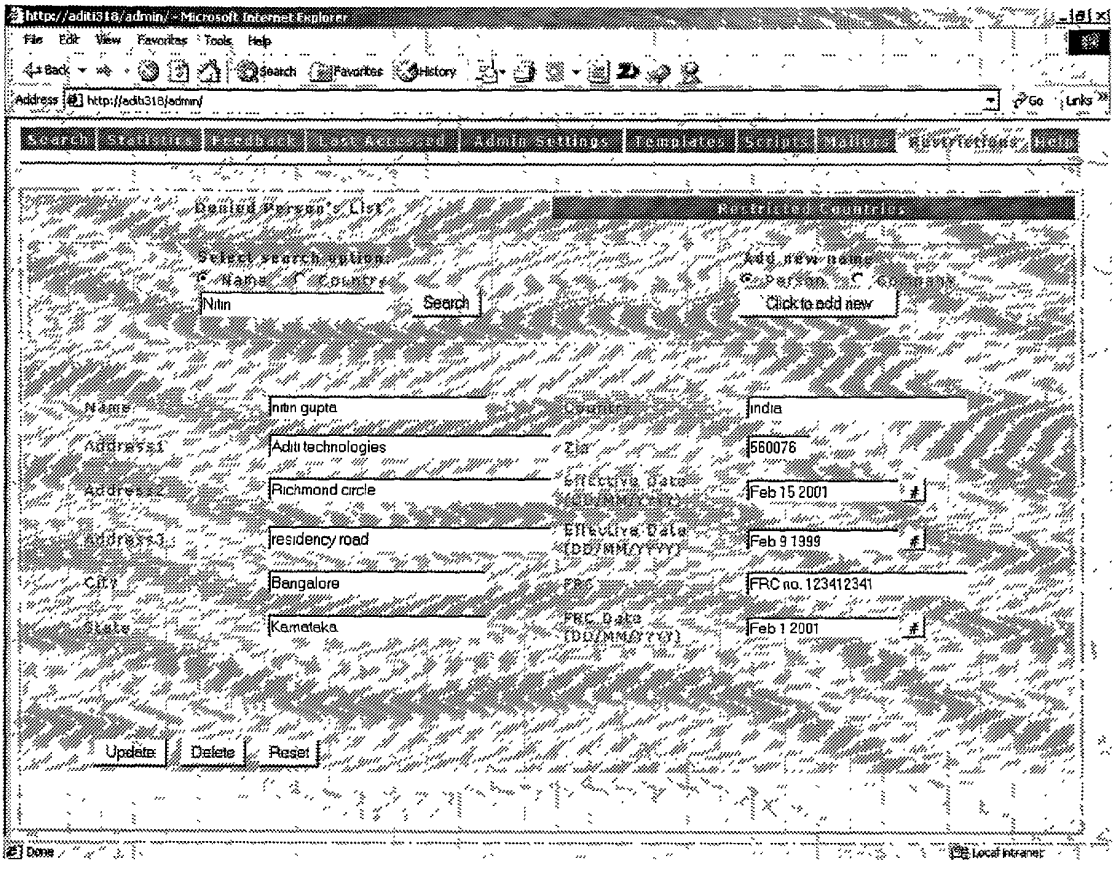
Description

This page contains all the mail contents. The admin will be able to update the mail content also.



4.2.10 Restrictions

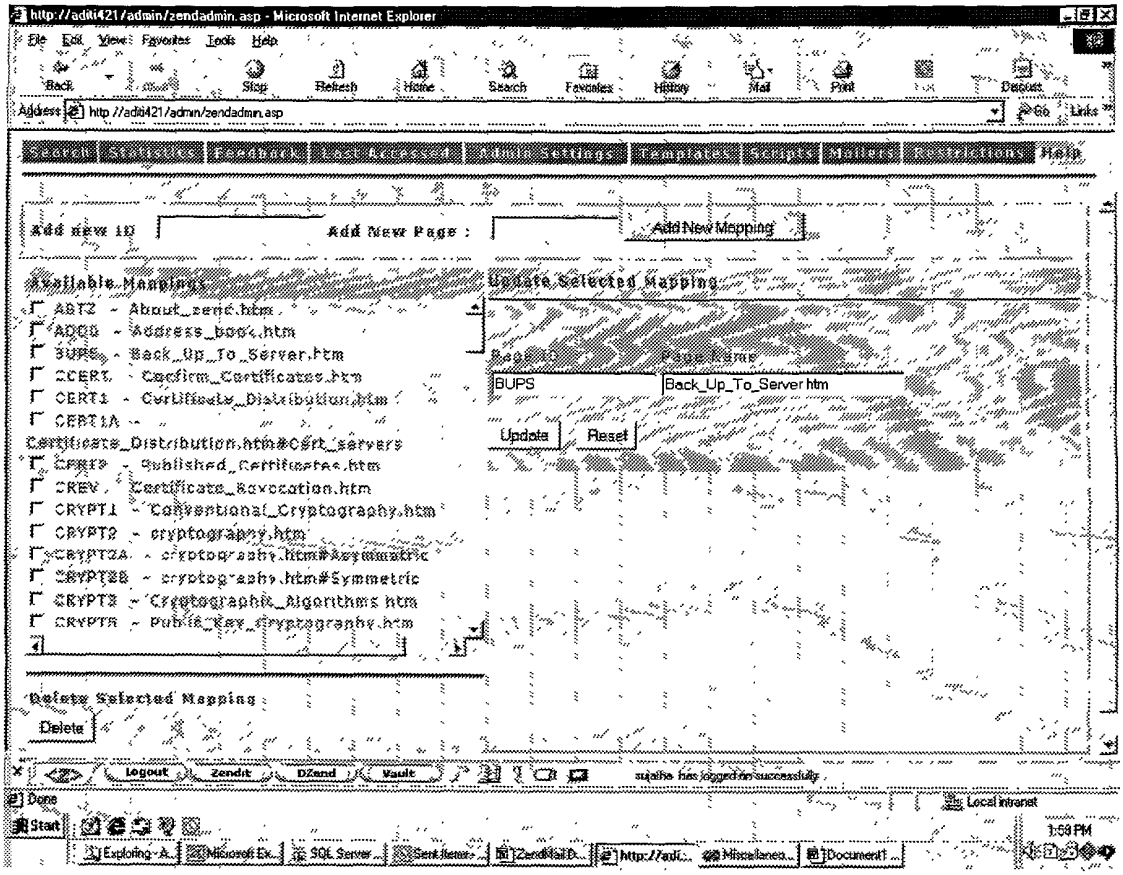
Description
This page is used to maintain a list of all the denied persons and companies. We can also add restricted countries to the list.



4.2.11 Help

Description

This page contains the help mappings between the identification code and the help page. The admin can add or update the mapping value.


http://aditi421/admin/zendadmin.asp - Microsoft Internet Explorer
Add New ID
Add New Page :
Add New Mapping
Available Mappings
ABT2 - About_zend.htm
ADDB - Address_book.htm
BUPS - Back_Up_To_Server.htm
CCERT - Confirm_Certificates.htm
CERT1 - Certificate_Distribution.htm
CERT1A -
Certificate_Distribution.htm#Cert_servers
CPUB - Published_Certificates.htm
CREV - Certificate_Revocation.htm
CRYPT1 - Conventional_Cryptography.htm
CRYPT2 - cryptography.htm
CRYPT2A - cryptography.htm#Asymmetric
CRYPT2B - cryptography.htm#Symmetric
CRYPT3 - Cryptographic_Algorithms.htm
CRYPT4 - Public_Key_Cryptography.htm
Update Selected Mapping
Page ID
Page Name
BUPS
Back_Up_To_Server.htm
Update
Reset
Delete Selected Mapping
Delete
Logout
Zendit
DZend
Vault
Local intranet
1:59 PM

4.3 Interface pages to SurfBoard.

These set of pages act as an interface to the SurfBoard. Any data needed by the SurfBoard will be requested to these pages. These pages in turn query the database and send the necessary data to the SurfBoard.

The following table depicts the various pages and the parameters sent and received. Each page is explained in detail after the table.

Interface pages between SurfBoard and Zendit Server

Request	Sent Fields	Return Fields
Login Request	strLogin strPwd flgTempKeyExists	INewMailStatus StrFirstName INewTemplateVer INewSurfBoardVer INewSurfBoardCriti cFactor CStatusHash CleanupTime flgStatus StrSQLServer MailFooter MailHeader
Request email IDs and Key IDs	strLogin strSQLServer strStatusHash	flgStatus iCount strEmailId[i] strKeyId[i] flgKeyRetStatus[i]
New Mail Status Request	strLogin strSQLServer strStatusHash	flgStatus flgMailStatus
Request Templates		flgStatus strTemplates strPassword
Request recipients' public keys	strEmailId[i] strLoginID	flgStatus iCount

Request	Sent Fields	Return Fields
	strSQLServer strStatusHash	strCert[i] flgStatus[i]
Backup Lock and Key	strLogin StatusHash strSQLServer strKeyPairData strEMailId strKeyId strName flgReplaceStatus strStatusHash	flgStatus
Restore Lock and Key	strLogin StatusHash strSQLServer	FlgStatus iCount strEmailId[i] KeyID[I] Name[I] FileBlob[I]
Public Key Upload	strLogin strPwd strEmailId strSQLServer strSQLNameForRe place strName strCert strKeyID strMailSystem strAlgo strLength dtCreatedate dtExpiryDate strStatusHash iReplace	flgStatus flgEmailExists strSQLServer strKeyId

Request	Sent Fields	Return Fields
Get User Personal Info	strEmailId strKeyId	FlgStatus <all public personal info> strKeyId strCertificate strEmailId dtExpiry
Public key delete	strLogin StatusHash strSQLServer strEmailId strKeyId	flgStatus
Temporary key pair creation request	arrEmailId(i)	flgStatus arrCertStatus(i) arrKeyID(i) arrPublicCert(i)
Mail(s) Zend notify	strLogin strSQLServer strEmailId[i] strTempEmailIDList strMailSystem strStatusHash	flgStatus
Reset mail status	strLogin strSQLServer strStatusHash	flgStatus
Download Temporary Keys	strLogin statusHash strSQLserver	flgStatus iCount strEmailID[i] KeyID[i] FileBlob[i]
Transparent Login into Vault	StrLoginID StrStatusHash StrSQLserver	

4.3.1 Login Request

When the user tries to login, a request is sent to the Server from the Surfboard.

Fields sent from the SurfBoard to the Server

strLogin - The login ID which the user gives.
strPwd - The password given by the user
flgTempKeyExists – If this flag is set, the server will return the number of Zendit Created keys existing for this loginID.

Fields returned from the Server to the Surfboard

flgStatus - A flag that indicates to the SurfBoard whether the login has succeeded or not.
strSQLServer - The SQL Server name in which the user's details are stored.
INewMailStatus – The flag to indicate if there is a new mail.
StrFirstName – First name of the user who has logged in.
INewTemplateVer – The latest version number of the templates.
INewSurfBoardVer- The latest version of surfboard
INewSurfBoardCriticFactor – The version of surboard in the user's machine should be above or equal to this number.
CStatusHash – hash(sessionID + date + time)
ICleanUpTime – The timelimit for cleaning up tblLoginStatus.
MailFooter – The content that goes as footer in all the mails zent.
MailHeader – The content that goes in the header in all the mails zent.

Server Action

The Server receives the login ID and the password from the SurfBoard and hashes the password using the fnHash method and checks for the validity of the hash. Login can fail for any one of the reasons below:

- SQL Server connection failure.
- The user has not yet confirmed his registration.
- Invalid login ID or password.

Accordingly, flgStatus will carry an error code to the SurfBoard.

Every time the user logs in, a hashing is done on the login ID and the Server will find the SQL Server in which the user's data is stored. The SQL Server's name is passed onto the Surfboard, so that for further fetching of data from the database, all the SQL Servers need not be searched.

If the Login is successful, hashing is done on the string formed by concatenating the session ID, date and time. This hash is sent to the client as well as stored in a database table tblLoginStatus along with login ID and the date - time stamp. This entry is made in the corresponding SQLServer in which the user's data is stored.

iNewTemplateVer is the version of the template which is currently in the database. This version number is compared with that of the SurfBoard registry entry and if they differ, a request is made from the SurfBoard to the Server to download the recent version of template. This comparison avoids the need for downloading the templates every time the user logs in.

A message indicating whether the user has successfully logged in will be displayed to the user. If login succeeds, a low priority thread is started to fetch the email IDs associated with the login ID.

4.3.2 Request email IDs and Key IDs

The SurfBoard makes a request to the Zendit Server to download the key IDs and corresponding email IDs for the login ID.

Fields sent from the SurfBoard to the Server

strLogin - The login ID of the user
strSQLServer - The SQL Server name in which the details of that user is stored.
StrStatusHash - The unique value which is passed on to surfboard while logging in the user.

Fields returned from the Server to the Surfboard

flgStatus - The flag that specifies whether the operation has succeeded or not.
iCount - Number of email IDs associated with that login ID.
strEmailID[i] - Array of email IDs.
strKeyID[i] - Array of key IDs.
flgKeyRetStatus[i] - Array indicating status for each email ID

Server Action

The server contacts the SQLServer and validates the statushash in tblLoginStatus. If the statushash is valid, the server fetches the list of email addresses associated with the user name from tblLoginEmailID. Then, each email address is hashed and the SQL server is identified. The Key IDs for the email address is got from tblConfirmedCerts from that server.

4.3.3 New Mail Status Request

The SurfBoard requests for the new mail status of the user who has logged in.

Fields sent from the SurfBoard to the Server

strLogin - The login ID of the user
strSQLServer - The SQL Server name in which the details of that user are stored.
StrStatusHash – The unique value that is passed on to surfboard while logging in the user.

Fields returned from the Server to the Surfboard

flgStatus - The flag which indicates whether the operation has succeeded or not.
flgMailStatus - The flag which indicates whether there is a new mail or not.

Server Action

This page accesses the tblRegisteredUsers for the new mail status flag for the login ID after verifying the status hash in the tblLoginStatus. This information is indicated to the SurfBoard through the flgMailStatus flag.

4.3.4 Request Templates

The SurfBoard sends a request for the new templates from the Zendit adminserver. This request is made from the SurfBoard if the template version with the SurfBoard is different from the one in the database.

Fields sent from the SurfBoard to the Server

Fields returned from the Server to the Surfboard

flgStatus - The flag indicates whether the operation has succeeded or not.
strTemplates - Set of mail system templates available in the database.
Password – The password used for encrypting the template blob.
Server Action

When the server receives the request, the template will be downloaded to the SurfBoard.

4.3.5 Request recipients' public keys (certificates)

If the user is zending mails, the SurfBoard searches the local key ring for the recipients' public keys. If the public key for few of the recipients doesn't exist in the local key ring, the SurfBoard will request those public keys from the Zendit server by sending the email IDs as parameters.

Fields sent from the SurfBoard to the Server

strEmailID[i] - Array of email IDs for which the certificates have to be fetched from server.
strLogin - The login ID of the user
strSQLServer - The SQL Server name in which the details of that user are stored.

StrStatusHash – The unique value that is passed on to surfboard while logging in the user.
Fields returned from the Server to the Surfboard

flgStatus - The flag that indicates whether the operation has succeeded or not.
iCount - Number of email IDs for which the public keys are to be fetched from database.
strCert[i] - Array of certificates
flgStatus[i] - Array of status flags which indicate the status of each certificate. (Whether it exists in the server or not)

Server Action

The status hash value is validated with the value in tblLoginStatus in the SQL Server. Then each email address is hashed and the SQL Server is identified. The tblConfirmedCerts table is queried for every email ID and stored in an array. This array is then sent to the SurfBoard.

4.3.6 Backup Lock and Key

The SurfBoard requests this page to backup the user's lock and key to the Zendit server.

Fields sent from the SurfBoard to the Server

strLogin - The user's login ID.
StatusHash - during login The hash value of sessionId+date string which was sent to surfboard
strSQLServer - The SQL Server name in which the details of that user are stored.
strEMailId - The email ID for which the user has created the lock and key.
strKeyPairData - The private, public key pair created for the email ID.
strKeyId - The key ID that has to be uploaded.
strName - Name of the user that will appear in his certificate.
flgReplaceStatus - Status flag indicating whether to replace the key or not.

Fields returned from the Server to the Surfboard

flgStatus - The flag which specifies whether the operation has succeeded or not. It also specifies whether lock and key for that email ID already exists in the database.

Server Action
The value of the status hash is validated with the value in the tblLoginStatus in SQLServer.
The Zendit server receives the email ID and lock and key that has to be backed up along with SQL Server name. The server queries the tblKeyUpload table whether a lock and key already exists for that email ID. If so, this information is passed onto the SurfBoard through the flgStatus, else the lock and key is backed up.
If a lock and key already exists for the email ID specified by the user, then the user will be given an option to replace the existing one. If the user chooses to replace, a request is made again.

4.3.7 Restore Lock and Key

The SurfBoard requests this page to restore the user's key pair from the Zendit server.

Fields sent from the SurfBoard to the Server

strLogin - The user's login ID.

StatusHash - The hash value of sessionId+date string whichwas sent to surfboard during login

strSQLServer - The SQL Server name in which the details of that user are stored.

Fields returned from the Server to the Surfboard

FlgStatus - The flag that indicates whether the operation has succeeded or not
Icount - Number of email ID - key pair of the user that are in the database.
StrEmailID[I] - Array of email IDs of the user for which key pairs are present in the database.
KeyID[I] - Array of Key IDs
Name[I] – Array of the corresponding names
FileBlob[I] – Array of the corresponding fileblobs
Note: There can be only one lock and key associated for an email ID but the user can backup more than one lock and key.

Server Action

The server receives the user's login ID and the SQL Server name in which the details of user are stored. The value of the status hash is validated with the value in the tblLoginStatus in SQLServer.
The server fetches all the email ID - key pairs of the user from the tblKeyUpload table and passes onto the SurfBoard.

4.3.8 Public Key Upload

The SurfBoard requests this page when the user uploads his public keys to the Zendit server database.

Fields sent from the SurfBoard to the Server

strLogin - The user's login ID.
strPwd - The user's password.
strSQLServer - The SQL Server name in which the details of that user are stored.
strEmailId - The email ID for which the public key has to be uploaded.
strName - The name of the user as it appears in the certificate.
strCert - The certificate which has to be uploaded.
strKeyID - The key ID.
iReplace - The flag which indicates whether to replace if a certificate already exists in the database for that email ID. (This flag will be null during the first request).
StrStatusHash – The unique value that is passed on to surfboard during login
StrMailSystem – Mail system of that email address.
StrAlgo
StrLength
dtCreatedate

dtExpiryDate
(properties of the certificate that is published)

Fields returned from the Server to the Surfboard

flgStatus - The flag which indicates whether the operation has succeeded or not.
flgEmailExists - The flag which indicates whether a certificate already exists in the database for that email ID.
strSQLServer - The SQL Server name in which the certificate details of the user are stored.
flgKeyID - Key ID

Server Action

The Zendit database is checked whether a certificate already exists for that email ID. If it exists, then server sets the "flgEmailExists" flag and passes it to the SurfBoard. The user will be given an option to replace the existing one or not. The SurfBoard in turn sets the "iReplace" flag and again sends the request to the server with all the other values. The server updates the database accordingly.

4.3.9 Get User Personal Info (this functionality is not used)

The SurfBoard requests this page to fetch the user information for a given email ID.

Fields sent from the SurfBoard to the Server

strKeyID - The associated key ID of the email ID.
strEmailID - The email ID for which the user wants to find the details.

Fields returned from the Server to the Surfboard

flgStatus - The flag which indicates whether the operation has succeeded or not.
strKeyID - The associated key ID of the email ID.
strCertificate - The certificate associated with the email ID.
strEmailID - The email ID.
<all public info> - This information depends on the choice of the user whose information is being requested.
dtExpiry - The certificate's date of expiry.

Server Action

The tblRegisteredUsers table is accessed for retrieving the user's information. TblUserProfile is accessed for retrieving the choice of the user.

4.3.10 Public key delete

The SurfBoard requests this page when the user deletes any of the uploaded keys.

Fields sent from the SurfBoard to the Server

strLogin - The user's login ID.
StatusHash - The hash value of sessionId+date string which was sent to surfboard during login

strSQLServer - The SQL Server name in which the details of that user are stored.

strEmailID - The email ID for which the user wants to delete the key ID.
strKeyID - The key ID which the user wants to delete.

Fields returned from the Server to the Surfboard

flgStatus - The flag which indicates whether the operation has succeeded or not.

Server Action

The EmailID is hashed to find the corresponding SQL server. A connection is made to that server and theCorresponding record is moved from tblConfirmedCerts table to the tblBadCerts table with a flag indicating "Delete" mode. The corresponding entry is also deleted from tblListEmailIDs and tblLoginEmailID .

4.3.11 Temporary key pair creation request

The SurfBoard sends a request to the server to create a interim locks and keys for those email IDs in the recipient's list for which the public keys are not available.

Fields sent from the SurfBoard to the Server

StrLoginID – Login ID of the user
StrSQLServer - The SQL Server name in which the details of that user are stored.
StrStatusHash – The unique value passed on to surfboard during login.
StrEmailIDList(i) - The email ID list for which the SurfBoard requests the certificates.

Fields returned from the Server to the Surfboard

flgStatus - The flag that indicates whether the operation has succeeded or not.

arrCertStatus(i) – The status flag which indicates the source of the certificate(confirmedcerts/confirmedtempkeys/tblTempKeyConfirmation/tblTempKeyPairPool)
arrKeyID(i) – the array of key Ids for the Email Ids.
arrPublicCert(i) – the array of certificates.

Server Action

The server checks whether a key ID already exists for that particular email ID in the tblConfirmedCerts or in tblConfirmedTempKeys or in tblTempKeyConfirmation table. If it exists, flgStatus will be sent to the SurfBoard accordingly. If the key ID doesn't exist , the mail will be sent using the new public key that was created(tblTempKeyPairPool) and sent by the server to the SurfBoard.

4.3.13 Mail(s) Zend notify

The SurfBoard requests this page to update the number of mails zent and the new mail status of all recipients.

Fields sent from the SurfBoard to the Server

strLogin - The user's login ID.
strSQLServer - The SQL Server name in which the details of that user are stored.

strMailSystem – The mail system from which the mail is sent.
strEmailID[i] - Array of email IDs in the "TO" list.
StrTempEmailIDList – List of EmailIDs for which the ZendSystem created KeyIDs.
StrStatusHash – Unique value passed on to surfboard when the user logs in.

Fields returned from the Server to the Surfboard

flgStatus - The flag which indicates whether the operation has succeeded or not.

Server Action

tblMailsZentLogin in strSQLserver will be updated for the corresponding LoginID and tblMailsZent in the admin server will be updated for the corresponding mailsystem and date.
Mails will be sent to all the email IDs in the strTempEmailList

4.3.14 Reset mail status

The SurfBoard requests this page to reset the new mail status flag once the user reads a mail using dezend.

Fields sent from the SurfBoard to the Server

strLogin - The user's login ID.
strSQLServer - The SQL Server name in which the details of that user are stored.
StrStatusHash – The unique value that is passed on to surfboard when the user logs in.

Fields returned from the Server to the Surfboard

flgStatus - The flag which indicates whether the operation has succeeded or not.

Server Action

The tblRegisteredUsers table is updated by resetting the new mail status flag for the given login ID.

4.3.15 Download TemporaryKeys

The SurfBoard requests this page to download the interim lock and keys for the corresponding LoginID from the Zendit server.

Fields sent from the SurfBoard to the Server

strLogin - The user's login ID.
StatusHash - The hash value of sessionId+date string whichwas sent to surfboard during login
strSQLServer - The SQL Server name in which the details of that user are stored.

Fields returned from the Server to the Surfboard

FlgStatus - The flag that indicates whether the operation has succeeded or not
Icount - Number of email IDs – interim lock and keys of the user that are in the database.
StrEmailID[I] - Array of email IDs of the user for which interim lock and keys are present in the database.
KeyID[I] - Array of Key IDs
FileBlob[I] – Array of the corresponding fileblobs

Server Action

The server receives the user's login ID and the SQL Server name in which the details of user are stored. The server fetches the email ID list for the corresponding LoginID from tblLoginEmailID for which interim lock and key exist. Then, keyID and the keypairblobs are retrieved for each emailID from tblConfirmedTempKeys from the corresponding server which is identified by hashing the email address.

4.3.16 Transparent Login to Vault

The surfboard requests this page to bring up the webvault inside the vault.

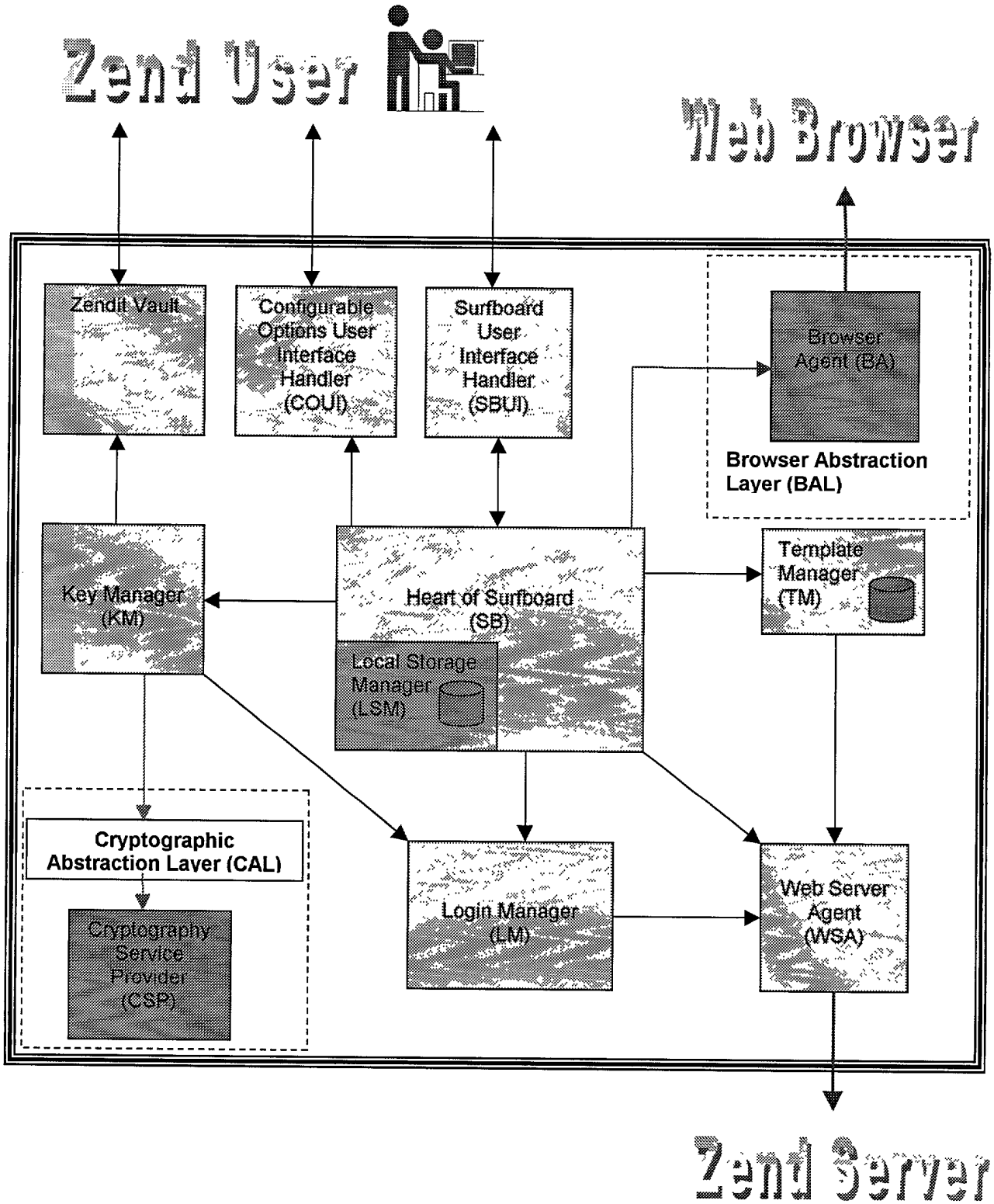
Fields sent from the SurfBoard to the Server

strLogin - The user's login ID.
StatusHash - The hash value of sessionId+date string which was sent to surfboard during login
strSQLServer - The SQL Server name in which the details of that user are stored.

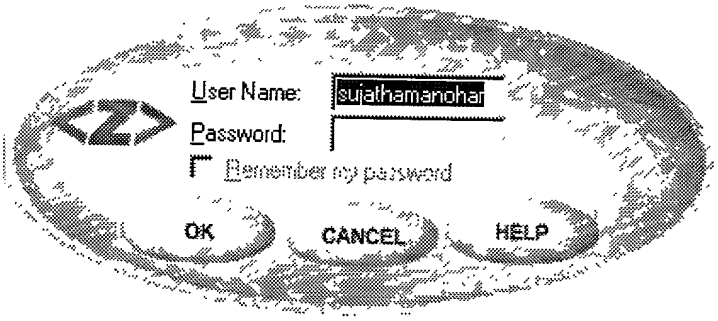
Server Action

The server checks the valididty of the statushash in tblLoginStatus. If it is valid, the web vault for the corresponding login is brought up inside the vault. Else, the login page is brought up.

5. SurfBoard Components



5.1 Login Manager



Login Name: The login name of the user as specified in the registration page at Zendit.com.

Password: The password of the above login name as specified in the registration page at Zendit.com.

Save Password: The option for the user to store the password in the protected storage on the local machine.

Description
Login Manager (LM) module takes care of logging in the user and maintaining the user's identity until the user exits the SurfBoard. LM logs in the user either by presenting the user with the login name/password dialog or in the case of auto login, it gets the user info from the protected storage. The LM maintains the state whether the user has logged in or not. Every other module that requires the user to be logged in queries the login manager and then performs the required action. The LM prompts the LSM to store the login name and password, which is requested by the web server agent to be passed, to communicate to the web server.

If the user selects the save password option, the Login Manager initiates the LSM to save the login name and password in the protected storage. The login name and password are encrypted and stored, which prevents the misuse of the registry values.

Assumptions
Once the user has logged in and opens another instance of Internet Explorer, the user information is taken from the login manager.
Similarly, as soon as the user logs out, user will be logged out of all the SurfBoard instances.

Limitations
In effect, only one user can logon to the SurfBoard on a system at a time.
SurfBoard stores only the last logged on user name and password (if save password is selected).

5.2 Browser Agent

Description
Browser Agent module interacts with the web browser (Internet Explorer) which hosts the SurfBoard. When the user is in the Compose, Reply, Forward page of a mailsystem which the zendit system supports and if the user initiates a "Zendit", the SurfBoard module requests the Browser Agent to fetch the values from the various fields like To, From, Cc, Bcc. The Browser Agent module gets the names of the various tags for the To, From, Cc and Bcc fields from the Template Manager. Using these tags the Browser Agent fetches the required values from the current page and passes them to the SurfBoard module.

5.4 Local Storage Manager (LSM)

Description
The local storage module manages all local data storage (includes: configurable SurfBoard options, saved user names and pass phrases, email system templates). All SurfBoard modules store and retrieve local data through the LSM, which provides them a single virtual view of Zendit's local storage. This allows us to maintain the flexibility to modify or migrate our local storage locations and methods.

5.5 Template Manager

Description
The Template Manager maintains the various information regarding the various mail systems' web pages. The template actually consists of the URL for Compose, Reply, Forward or Read page along with the various other information as specified in the database. This information is used by the Browser Agent to get the values from the To, From, Cc and Bcc fields. When the SurfBoard is started for the first time, it downloads the templates from the web server and stores in the local machine. Whenever a new version of the templates is put on the web server, the Template Manager downloads the latest version of Templates by requesting the Web Server Agent.

Assumptions:
The Administrator maintains the correct templates of the various mail systems' web pages.

5.6 Web Server Agent

Description

The Web Server Agent takes care of all the communication between the SurfBoard and the web server. Whenever the data is needed from the server, Web Server Agent communicates with the server by sending appropriate request.The data is received in the form of customised XML tags.

5.7 SurfBoard Module

Description

The SurfBoard module coordinates with all the other modules and processes the user's requests. As soon as the user selects SurfBoard from View / Explorer Bar (Menu of Internet Explorer), the SurfBoard module instantiates the User Interface. The SurfBoard module logs in the user with the help of the Login Manager. When the web server authenticates the user and sends the result, it also sends the latest template version to the SurfBoard. The SurfBoard module passes this template version information to the Template Manager, and the necessary action is taken. The SurfBoard module starts a low priority thread, which downloads the email IDs and the corresponding key IDs of the current user by requesting the Web Server Agent. When the user initiates a "Zendit" request, the following actions occur:

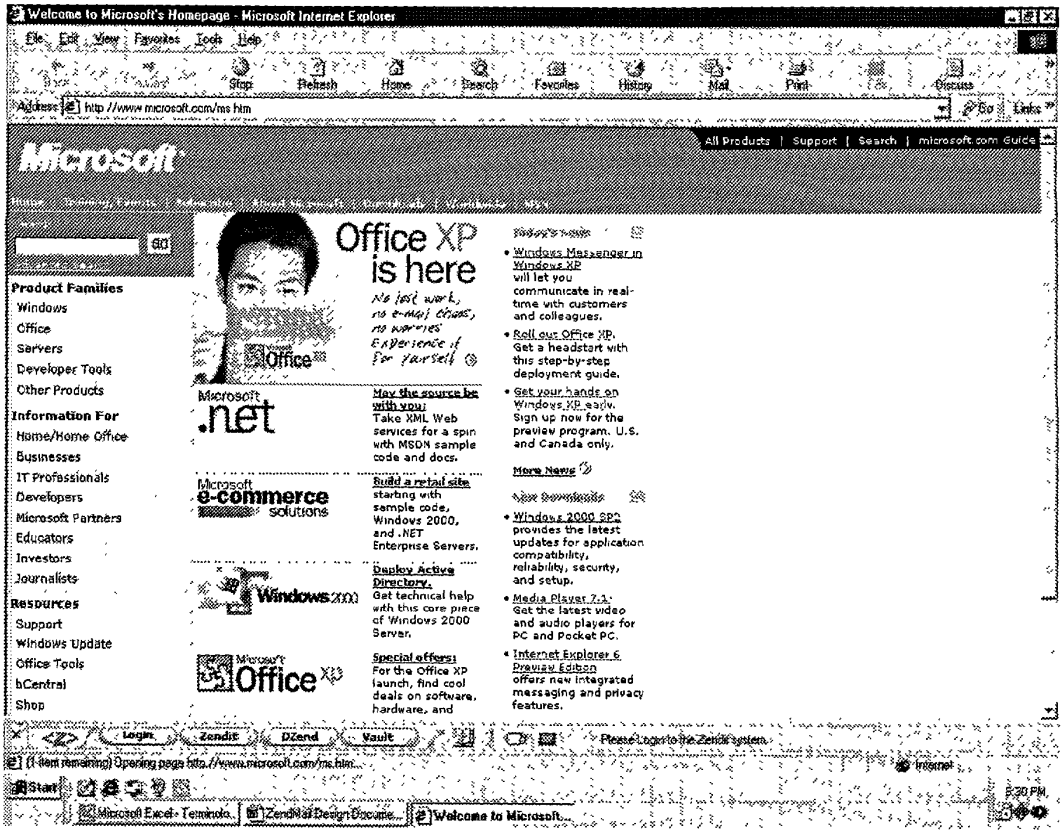
- The SurfBoard module requests the Browser Agent to get the values from the To, From, Cc and Bcc fields.
- It checks for the public keys corresponding to the mail IDs in the To, Cc and Bcc fields. This checking is done in the following order:
    - Checks in the local key ring.
    - If the public key doesn't exist in the local key ring, it searches for the public key in the Zendit server database and if it finds a match, it downloads the public key to the local key ring.
    - If the public key could not be found in the local key ring or in the Zendit server database, it asks the user if a interim lock and key has to be created for that email ID on the server.
        - If yes, a call is made to WSA to create interim lock and key.
        - If no, that email ID is removed from the To, Cc or Bcc list.

The SurfBoard module contacts the Key Manager for all key related activities.

5.8 SurfBoard UI

Description

SurfBoard UI module is the interface between the user and all the Zendit functionality. This handles the SurfBoard that gets attached to the Internet Explorer browser window and allows the user to encrypt mails using any supported mail system.



Assumptions

When the user zendit(encrypts) an email, the mail is signed, compressed and encrypted. The encrypted text is then converted into an ASCII armor, which replaces the original message body in the email.

Limitations

- When a user Zends an email, the new mail status of all the recipients is updated. This is unrelated to actual delivery or receipt of the email since those are handled by the individual mail systems and are outside our control.
- When the user receives an email with digital signature, the digital signature cannot be verified if the certificate used for signing is not available either in the local key ring or on the Zendit server.
- User will be able to re-encrypt an email for a different user(s).
- The file that goes as an attachment in a mail will not be automatically encrypted or decrypted The user cannot be intimated about attachments accompanying a mail.The system would not be able to resolve nick names to the appropriate email ID while Zending mails.
- Once user decrypts an email, it will be temporarily decrypted and displayed on the web browser. Since the email is stored in the mail system the user cannot save a decrypted copy of the email. Hence, the user must decrypt the email every time it is to be viewed.

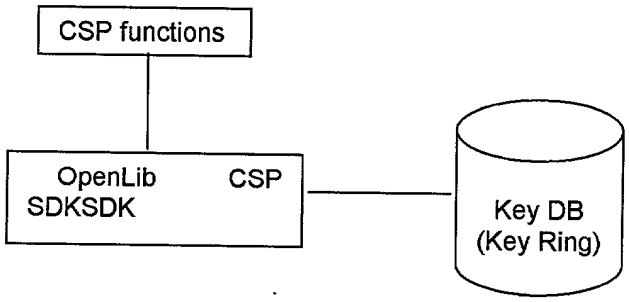
5.9 CSP

Description

Cryptographic Service Providers' (CSP) are independent modules that provide the cryptographic services to the application. It is written to be completely independent of any particular encryption algorithm, so that the Zendit SurfBoard application may, in future, work with a variety of CSPs. In practice, however, some modifications may be required to fully support multiple CSPs. An interface between the Zendit application and OpenLib CSP SDK will be implemented to maintain the application's independence from the underlying encryption technology. This makes replacing the OpenLib CSP SDK by another cryptography system in future relatively easy.

CSP functions can be categorized into the following sections:

- Key Management
- Encryption/Decryption functions.
- General functions.



Assumptions

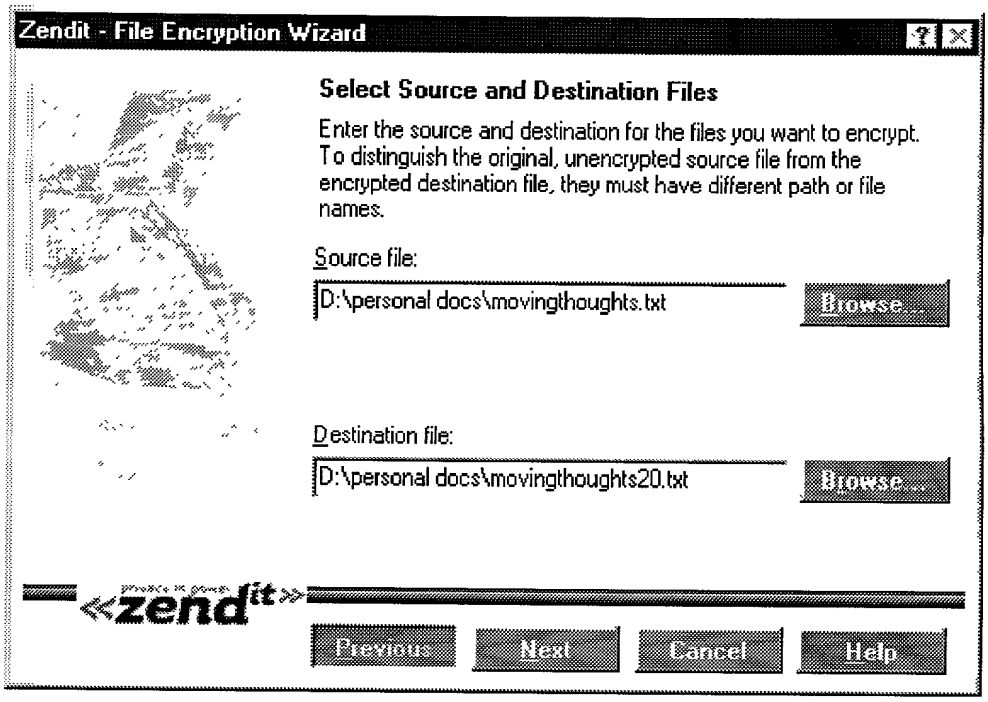
The OpenLib CSP is viewed as a complete, reliable package and will be used as is. Hence this will not be tested.

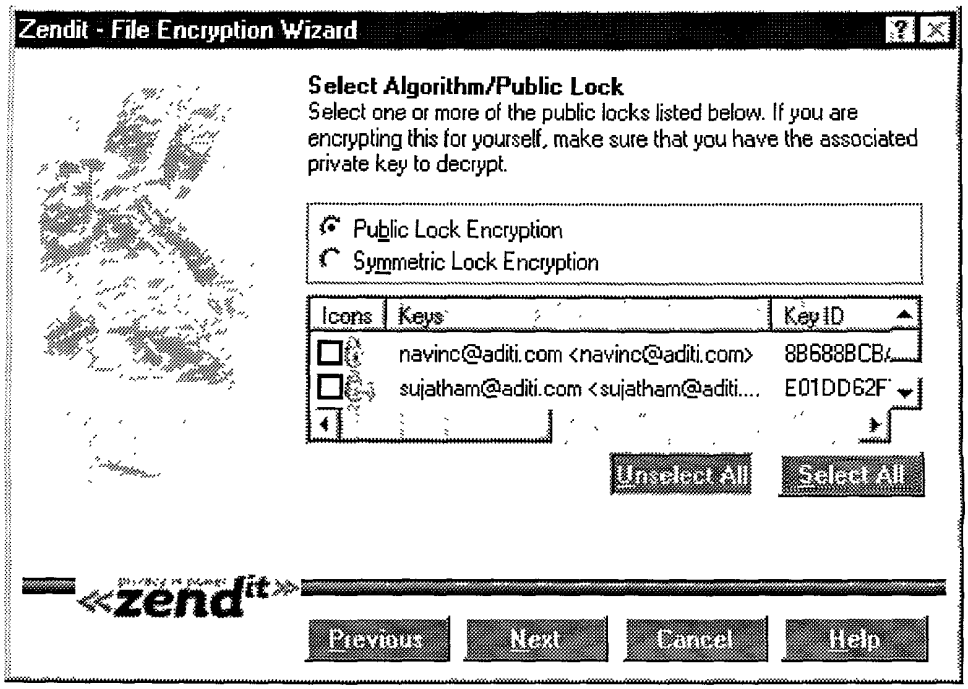
5.10 Text or File Encryption

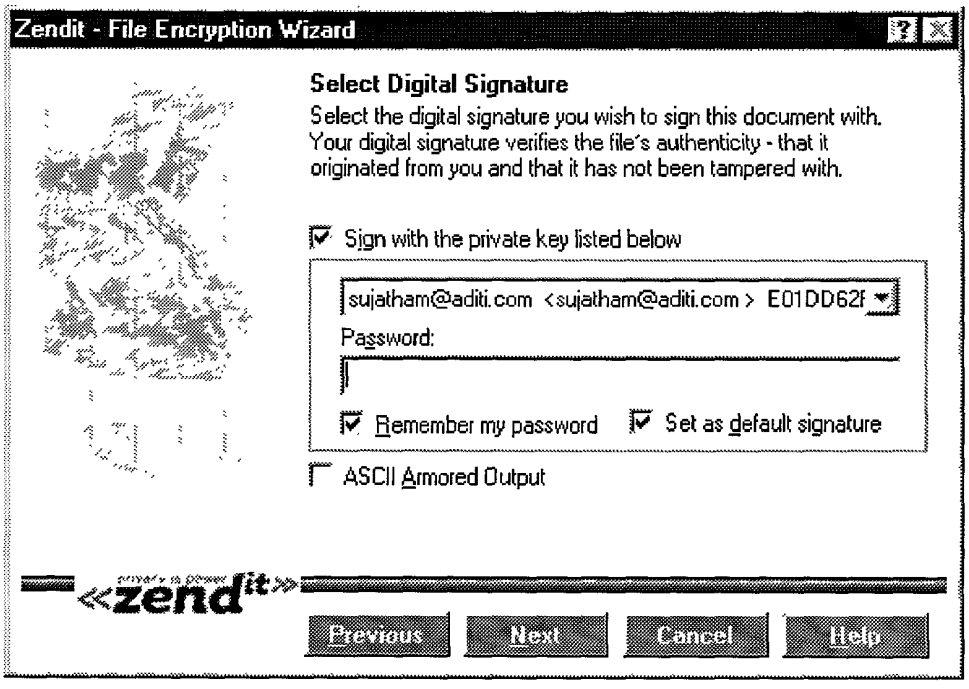
Description

The user can encrypt text or a file by selecting an appropriate menu item. If the user has logged in, either symmetric key encryption or public key encryption can be used. If the user has not logged in, only symmetric key encryption can be used (i.e. the option of public key encryption is disabled).

5.10.1 File encryption using public key







Encryption Type: The encryption algorithm to be used.

Choose Public Key: The user has to choose one Public key among the keys present in the key ring for public key encryption.

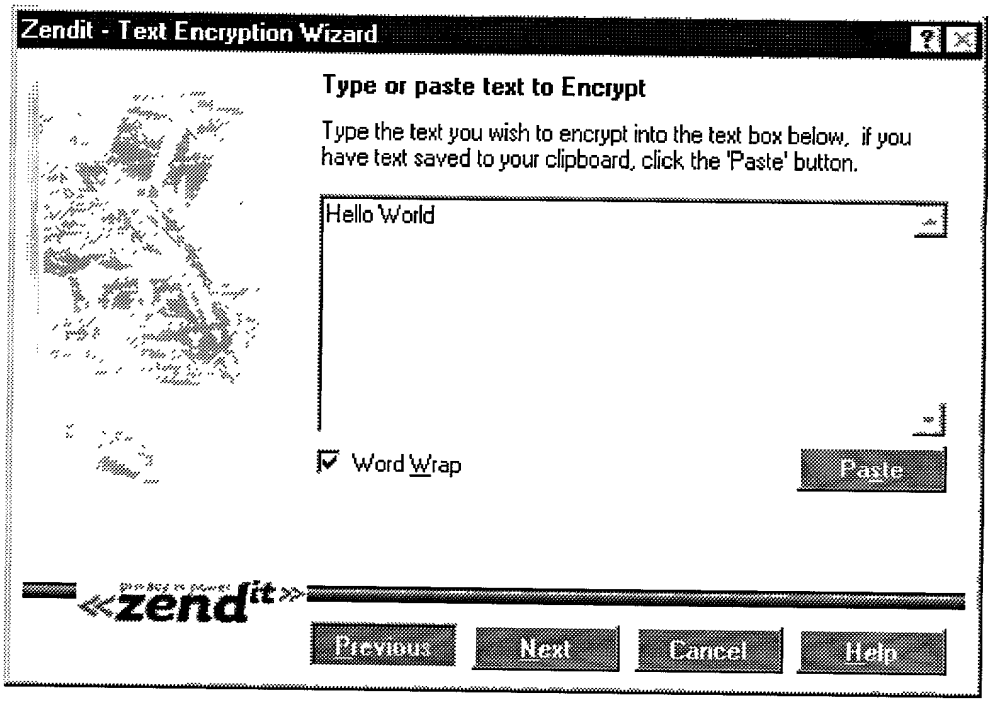
File Name: The actual file to be encrypted.

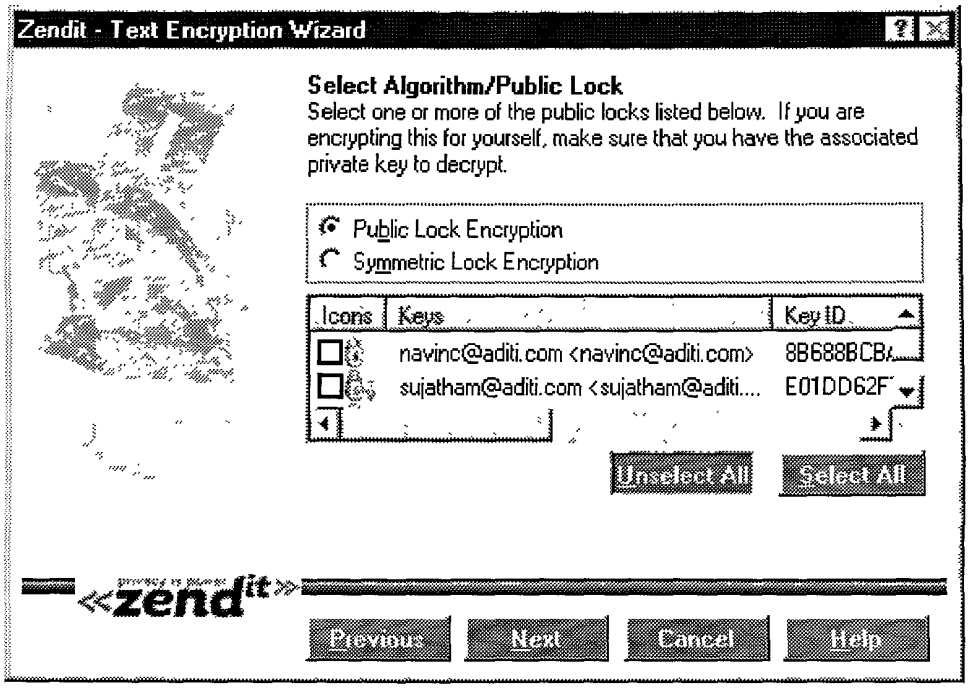
Encrypt: Encrypt the file specified with the selected public key.

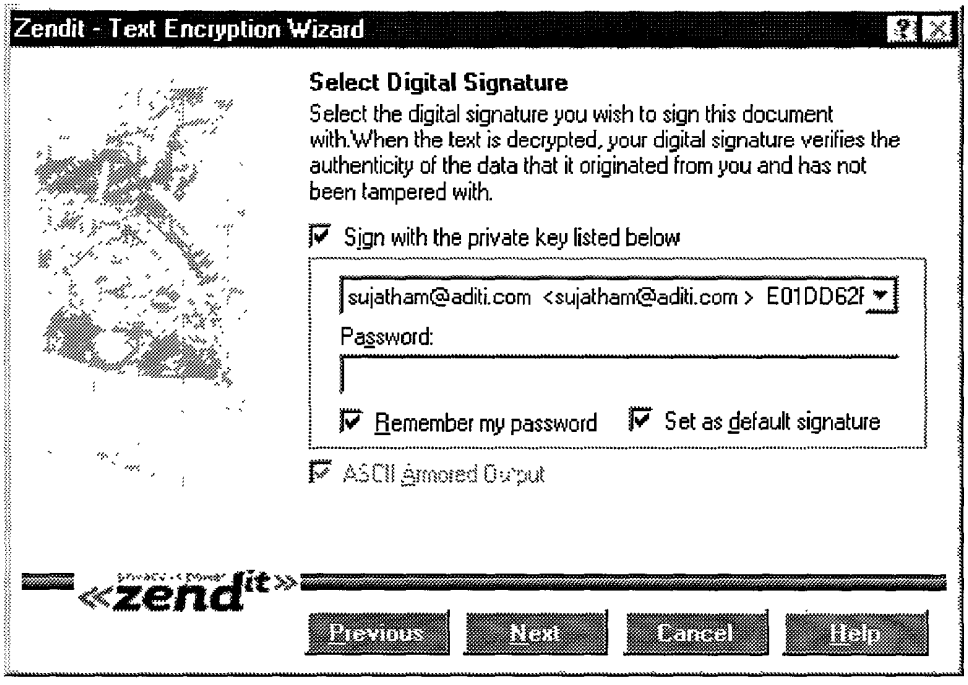
Select Digital Signature: The user is provided with an option to sign the encrypted file. He can choose the private key to sign.

Description
The specified file is encrypted using the public key selected and the resulting file is stored in the location specified by the user.

5.10.2 Text encryption based on public key







Encryption Type: The encryption algorithm to be used.

Choose Public Key: The user has to choose one Public key among the keys present in the key ring for public key encryption.

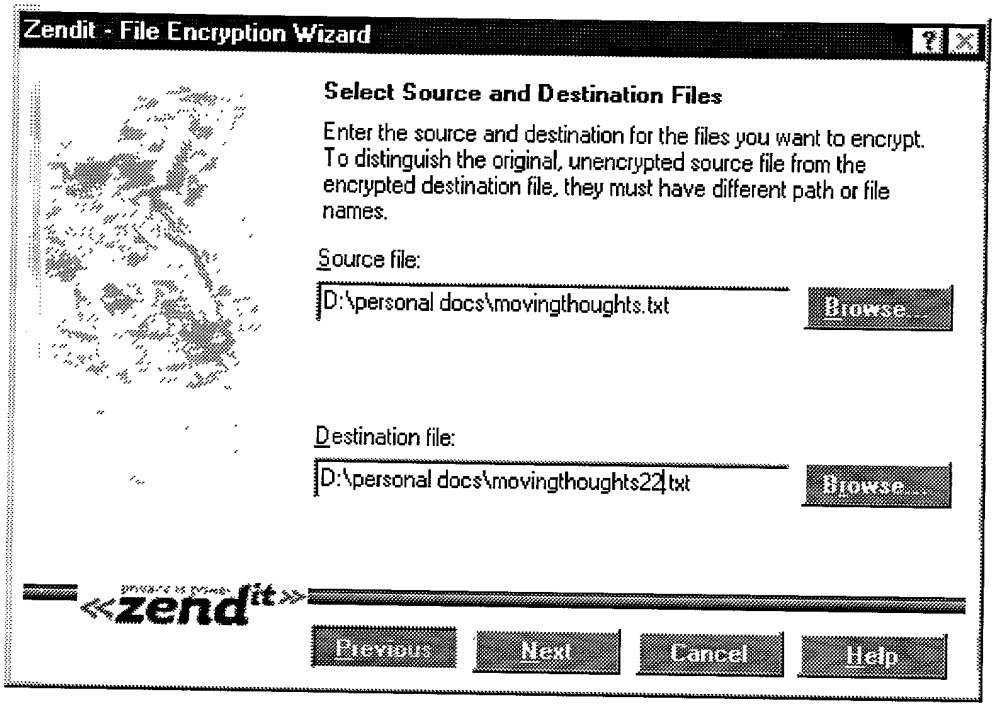
Text To Encrypt: The user has to enter the text to be encrypted.

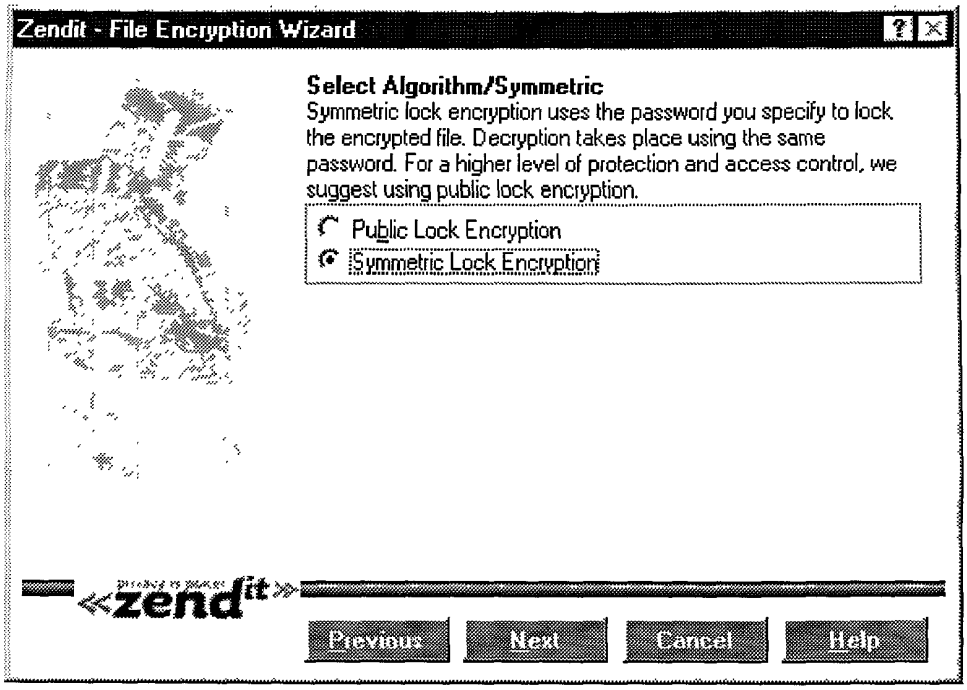
Encrypt: Encrypt the text with the selected key.

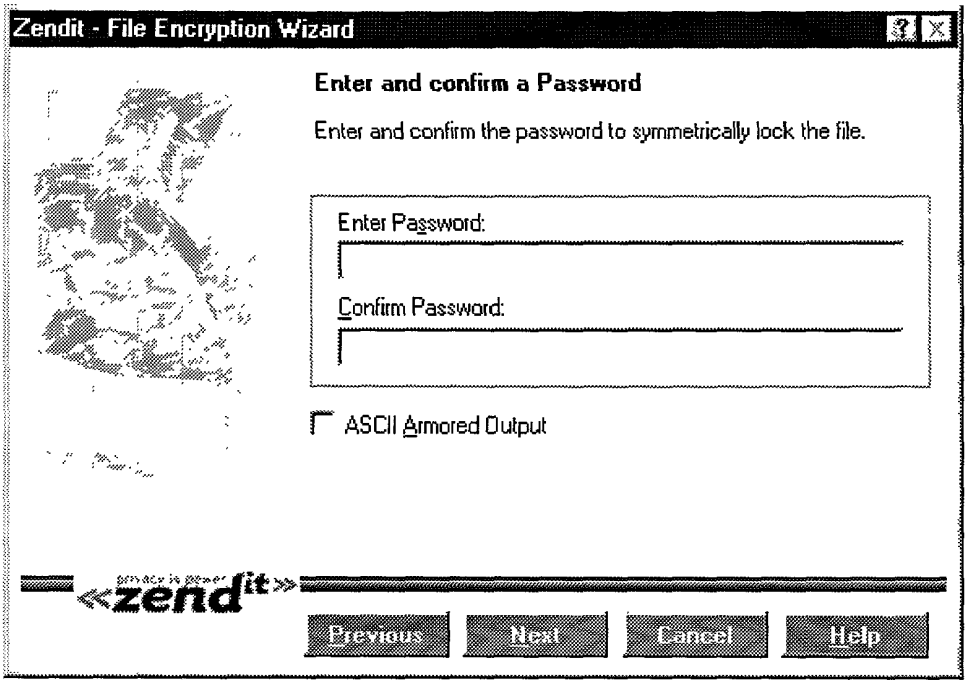
Select Digital Signature: The user is provided with an option to sign the encrypted text. He can choose the private key to sign.

Description
The text entered is encrypted using the public key specified and the result is displayed.

5.10.3 File encryption based on symmetric key







Encryption Type: The encryption algorithm to be used.

Symmetric Key Algorithm: The user selects the algorithm for symmetric key encryption.

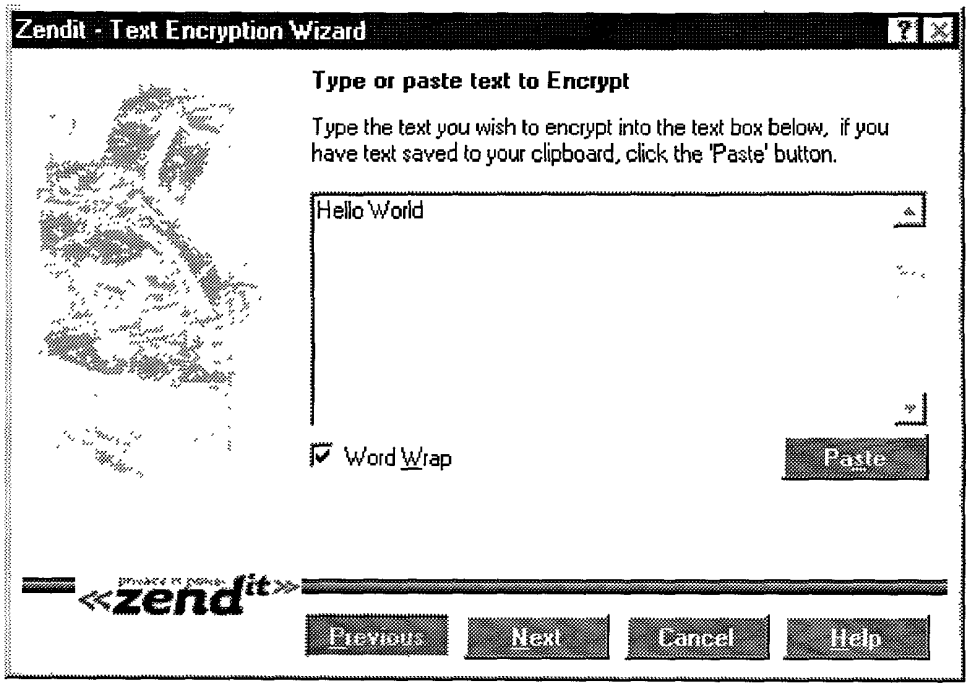
Symmetric Key Password: The user has to enter the password for the symmetric key encryption.

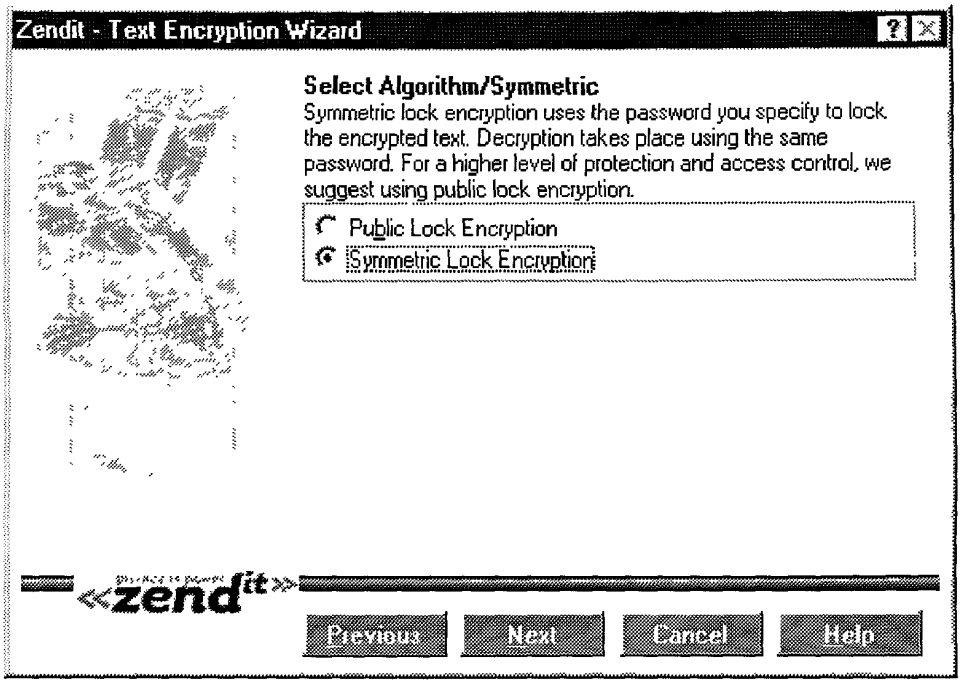
File Name: The file to be encrypted.

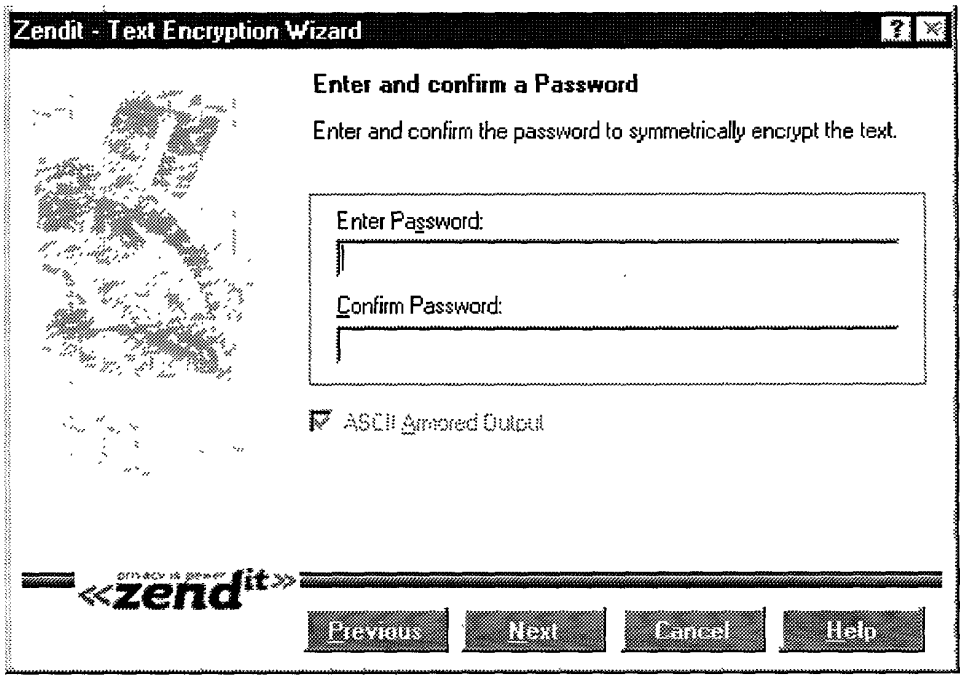
Encrypt: Encrypt the file specified with the algorithm selected.

Description
The specified file is encrypted using the selected symmetric key algorithm and the resulting file is stored in the location specified by the user.

5.10.4 Text encryption based on symmetric key







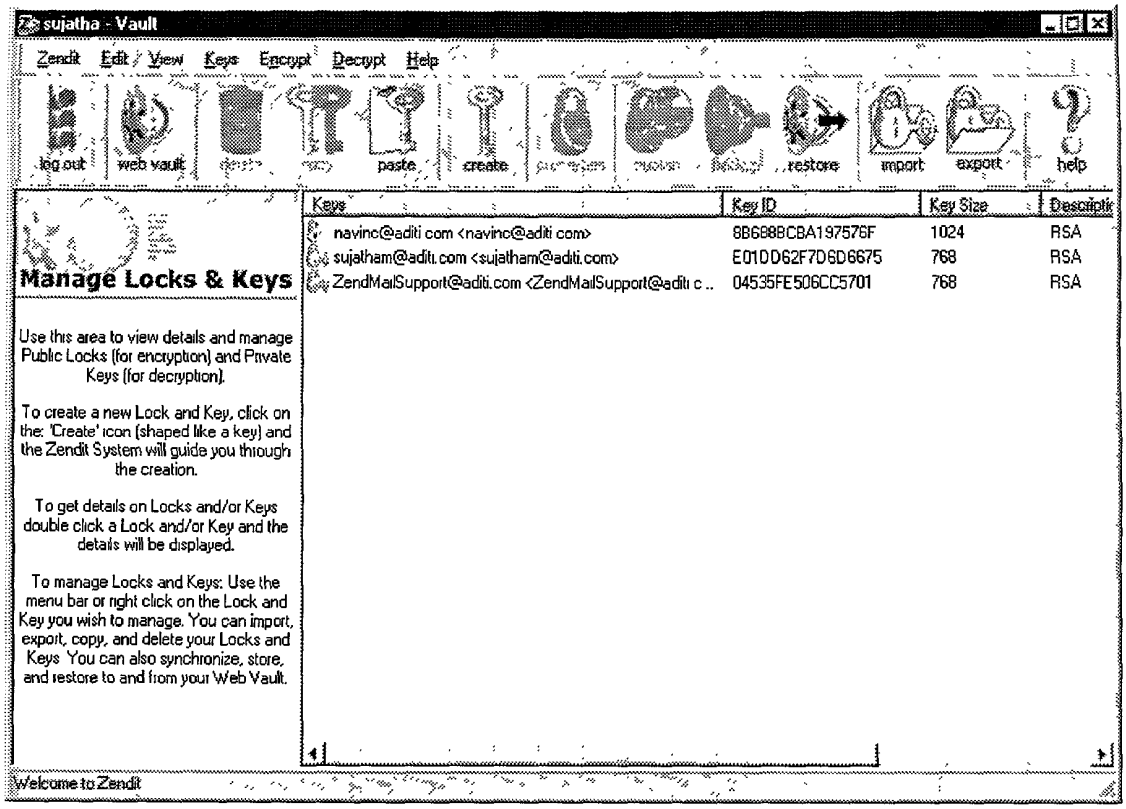
Encryption Type: The encryption algorithm to be.

Symmetric Key Algorithm: The user selects the algorithm for symmetric key encryption.

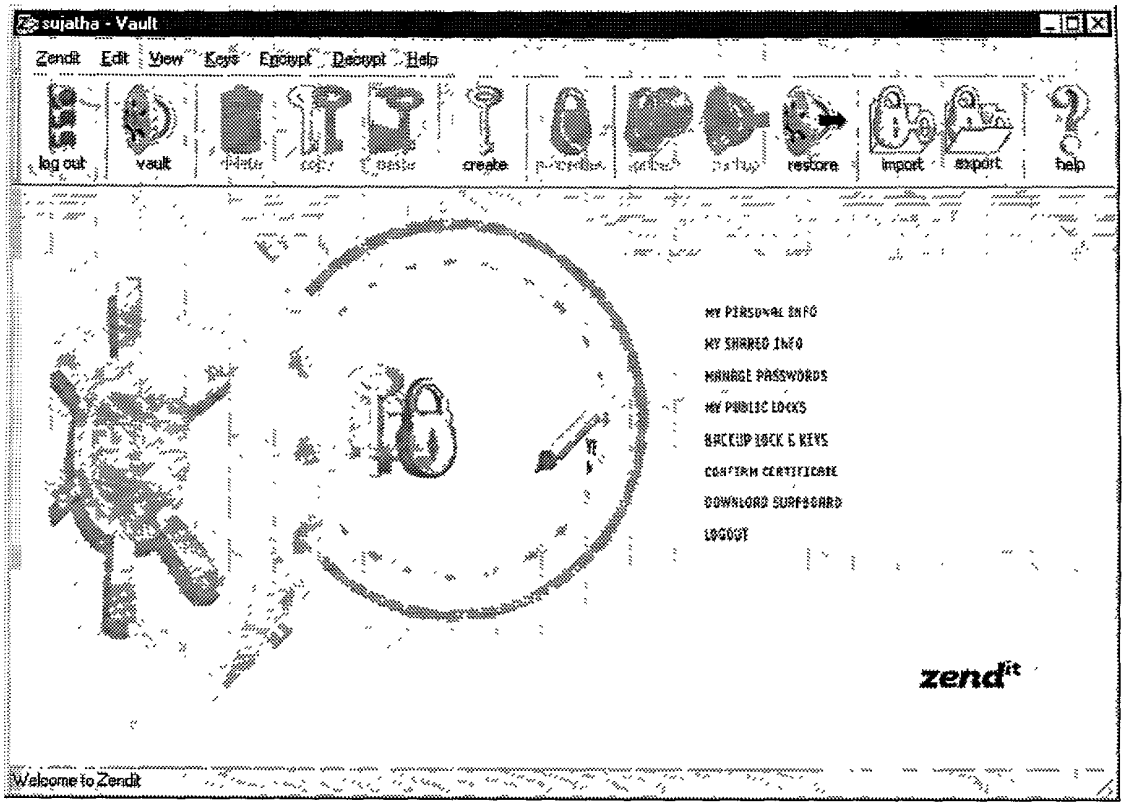
Symmetric Key Password: The user has to enter the password for the symmetric key encryption.

Text To Encrypt: The user has to enter the text to be encrypted.

Zendit Vault(Details)



View Web Vault :



Zendit Vault shows the details of all the keys, available in the Vault

Functioning

This module passes the user requests to the 'Zendit vault' module for required action.

As shown the user can choose to do one of the following:

- Create a new key
- Delete a key from the zendit vault
- Import a key from a local file into the zendit vault
- Export a key to a local file
- Backup Locks and Keys to the web vault
- Restore locks and keys to the web vault
- Publish public lock
- view the certificate associated with the key
- Search for a public lock and import
- Synchronize with web vault

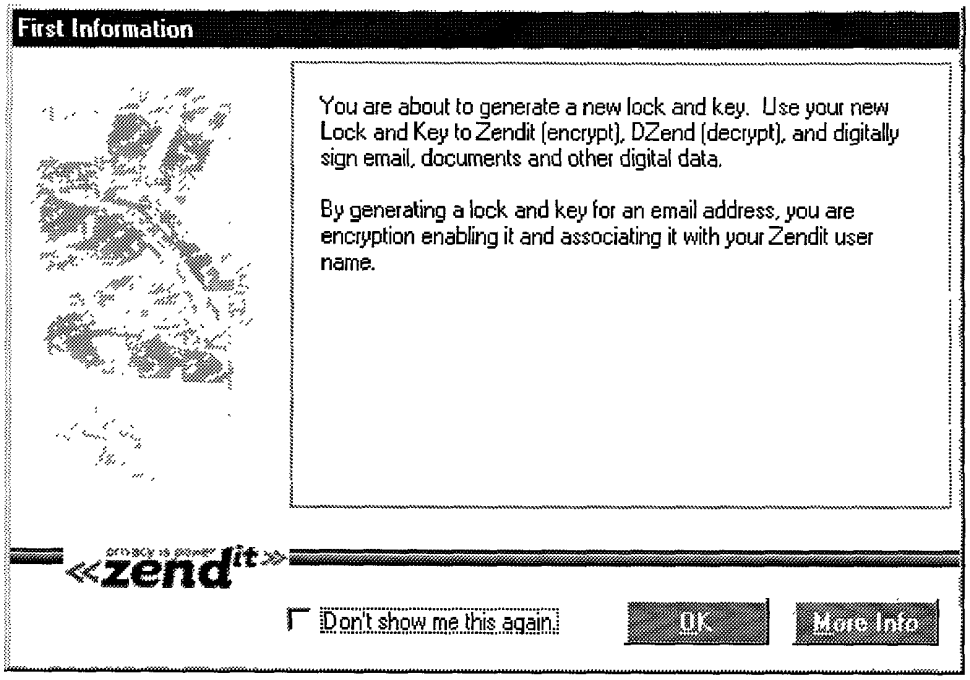
5.12.1 Create a new key

When the user chooses this option, the Zendit vault brings up a wizard. This wizard takes necessary input from the user. The wizard actions are explained in detail below.

Key Creation Wizard

Page 1: Introduction

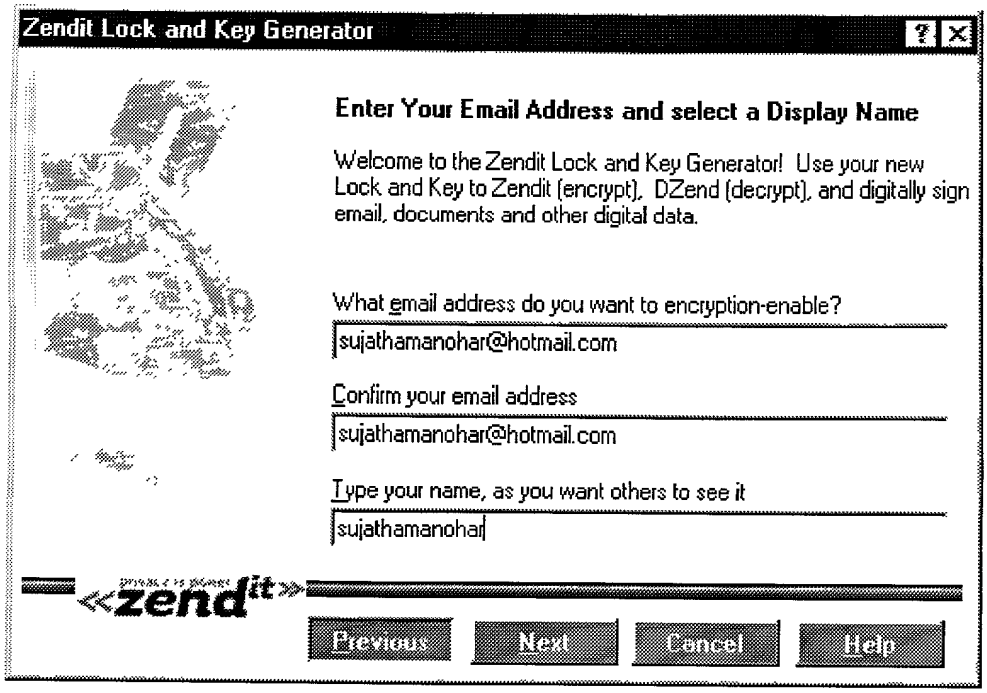
The first page in the wizard explains the need for a lock and key.



Page2: Name and email

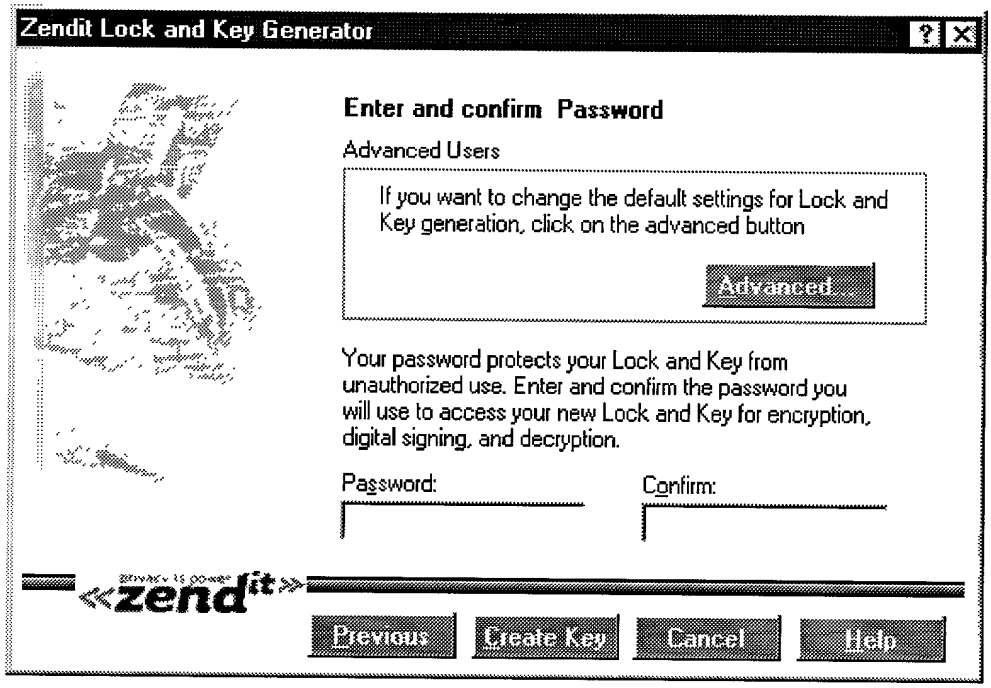
Name: The name with which the key will be associated.

Email ID: The email address with which the key will be associated.



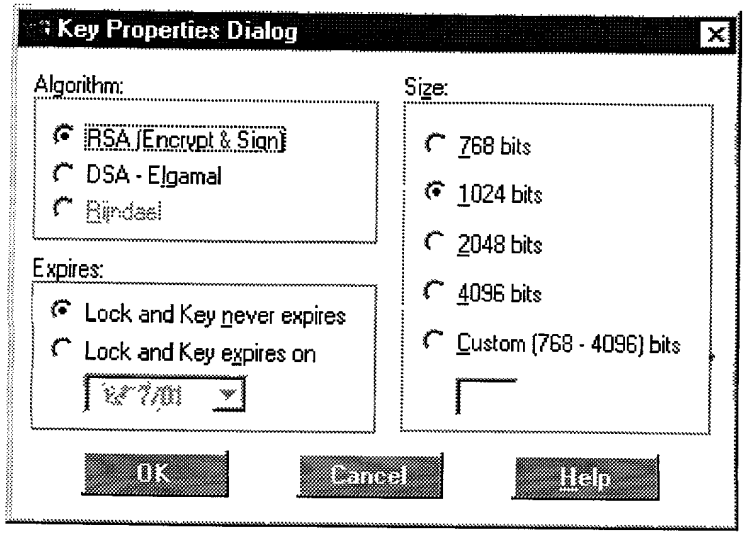
Page3: Password for the private key and the settings for the key
Password: The password that protects the user's private key.

Confirm: The user re-enters his password here for confirmation.

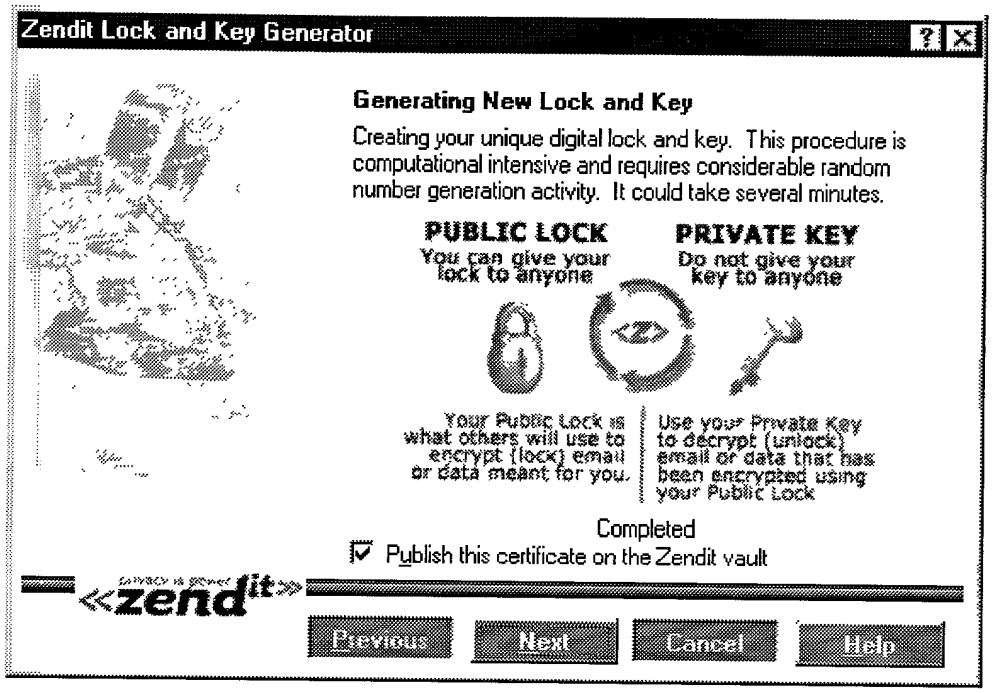


The user has an option to change the settings for the key if he goes through "Advanced" window.

If the user clicks on "Advanced" the following dialog is brought up where he can change the algorithm/expiry date/size of the key.

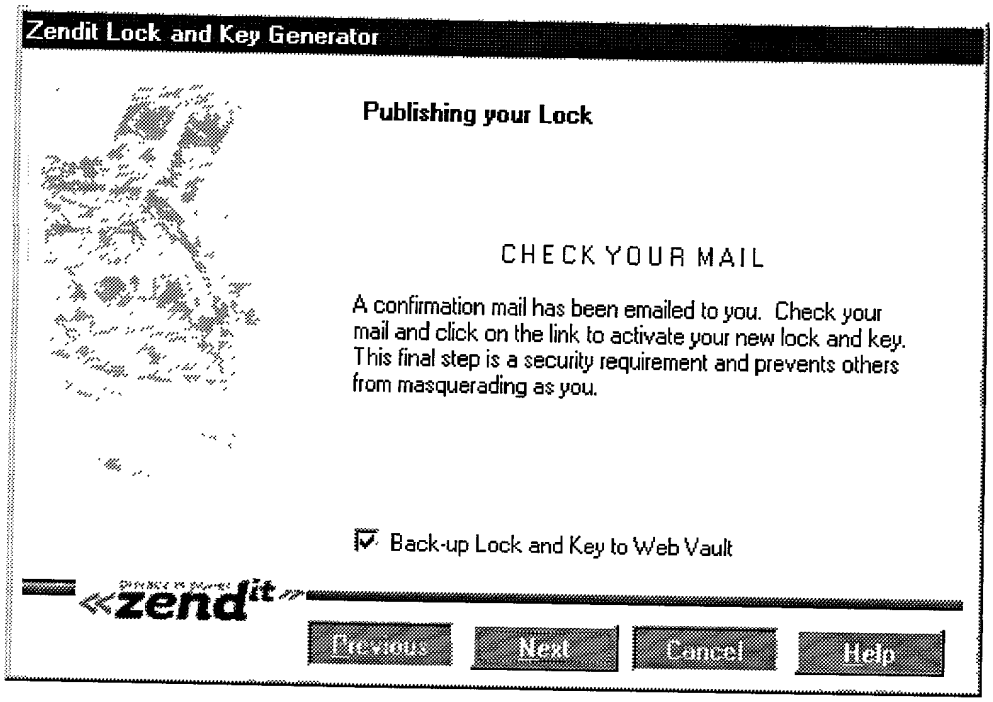


Page4: Option to publish the certificate on the Zendit vault.

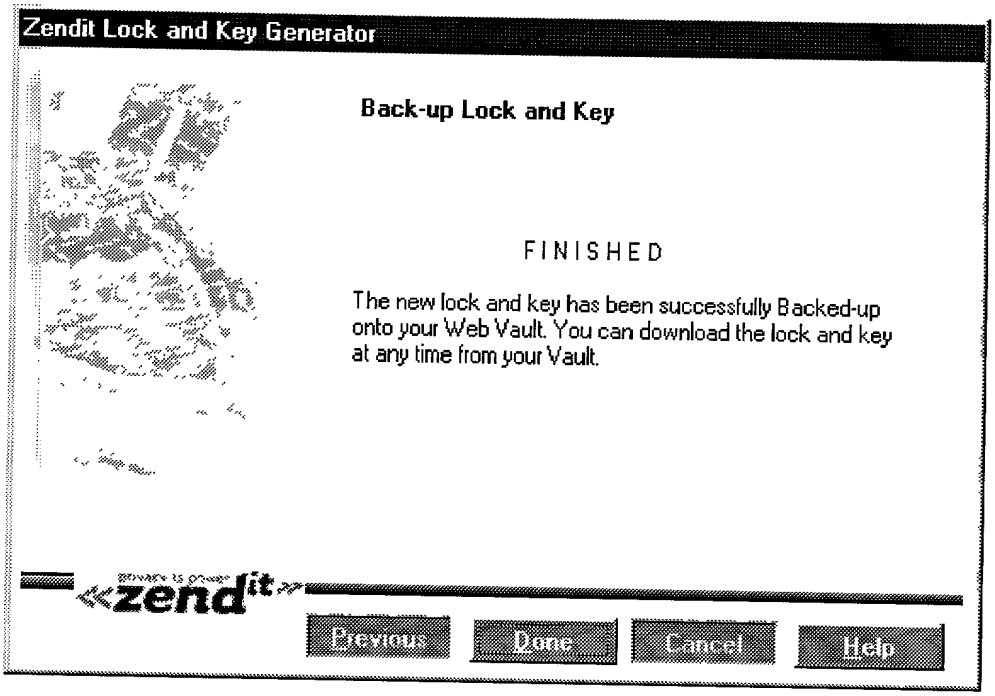


If the user does not select this option, then this public lock will not be available for others to encrypt the mail for this email address.

Page5: Option to backup lock and key to Web vault



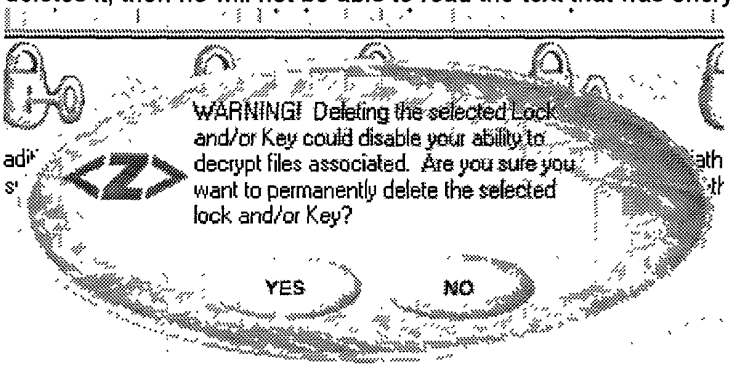
Page6: Backup Success

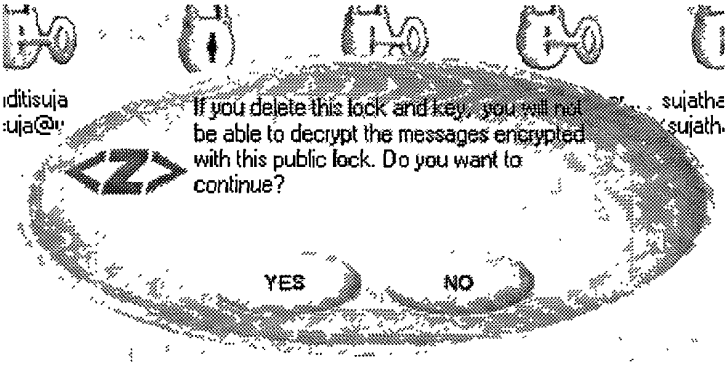


5.12.2 Delete a key from the zendit vault

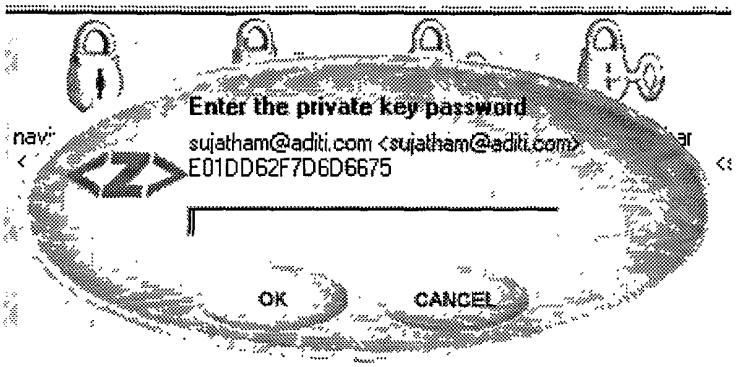
Description

When the user chooses a key from "vault' , a warning will be displayed to the user that if he deletes it, then he will not be able to read the text that was encrypted with this public key.

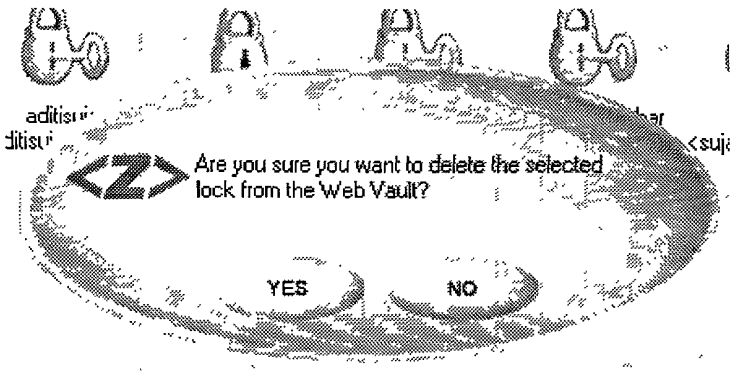




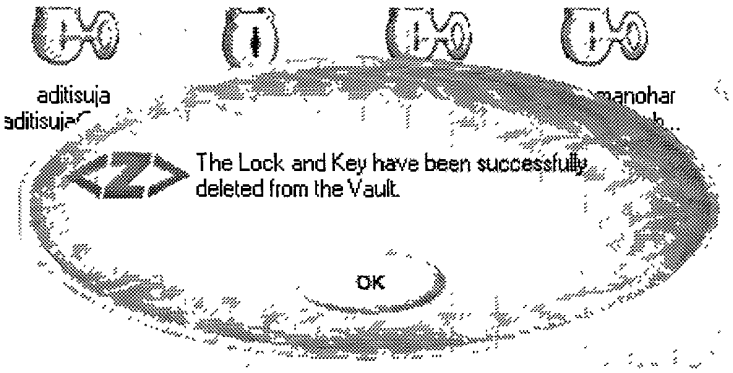
If the user still wants to delete the key, the user needs to submit the password for his private key.



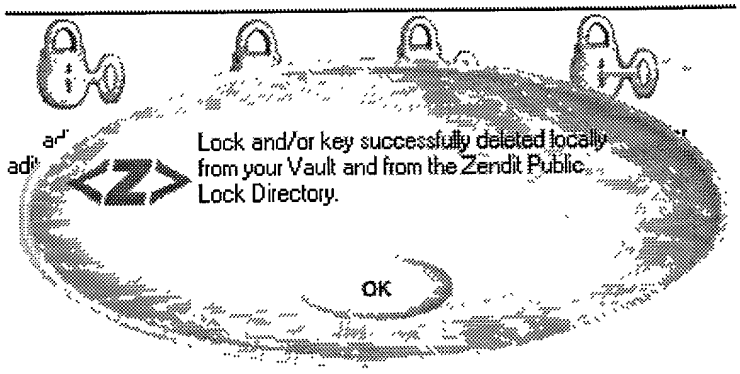
If the user enters the correct password, he will be given an option whether to delete from the web vault also.



If "No", then the key is deleted only from the Zendit vault.



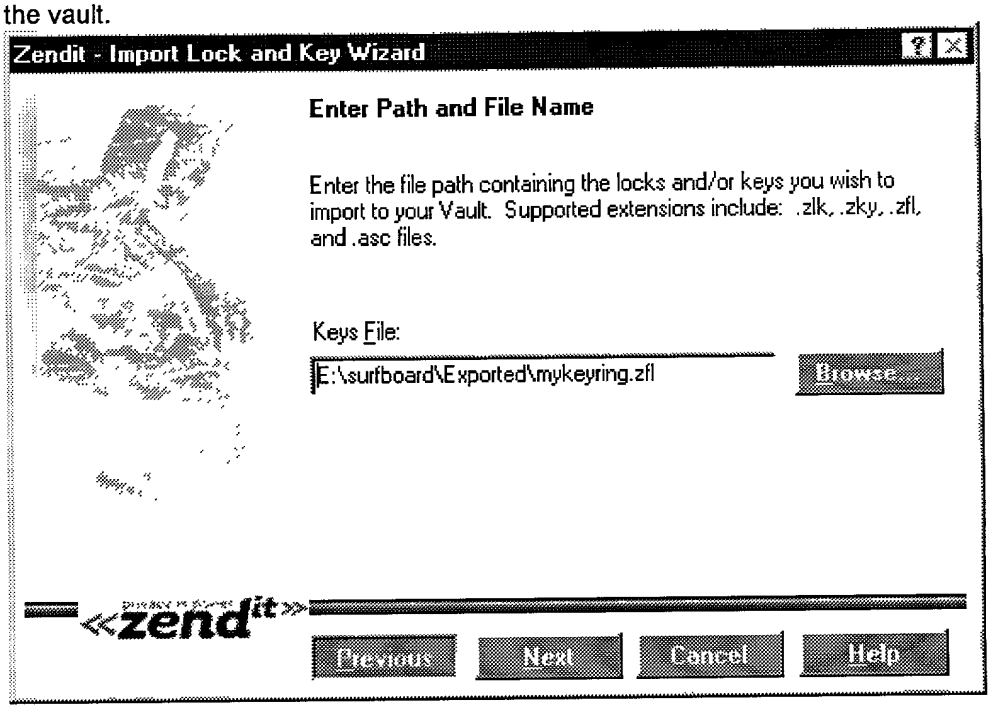
If "Yes", the key is deleted from the Vault and also the Web Vault

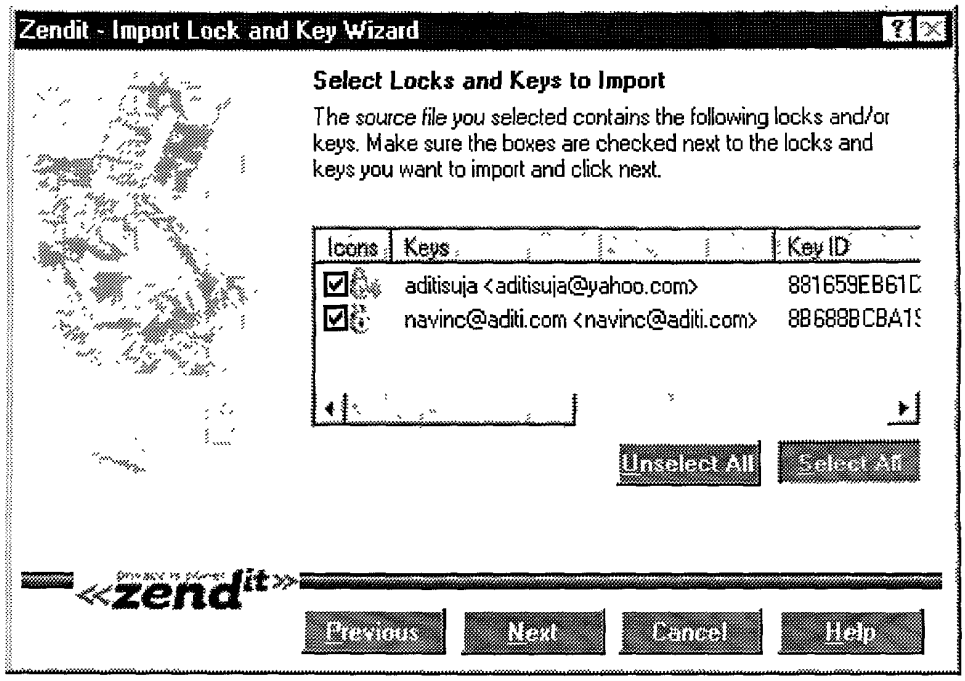


5.12. 3 Import a key from a local file into the zendit vault

Description

The user is prompted to select the key file to be imported. The selected file is then imported into the vault.

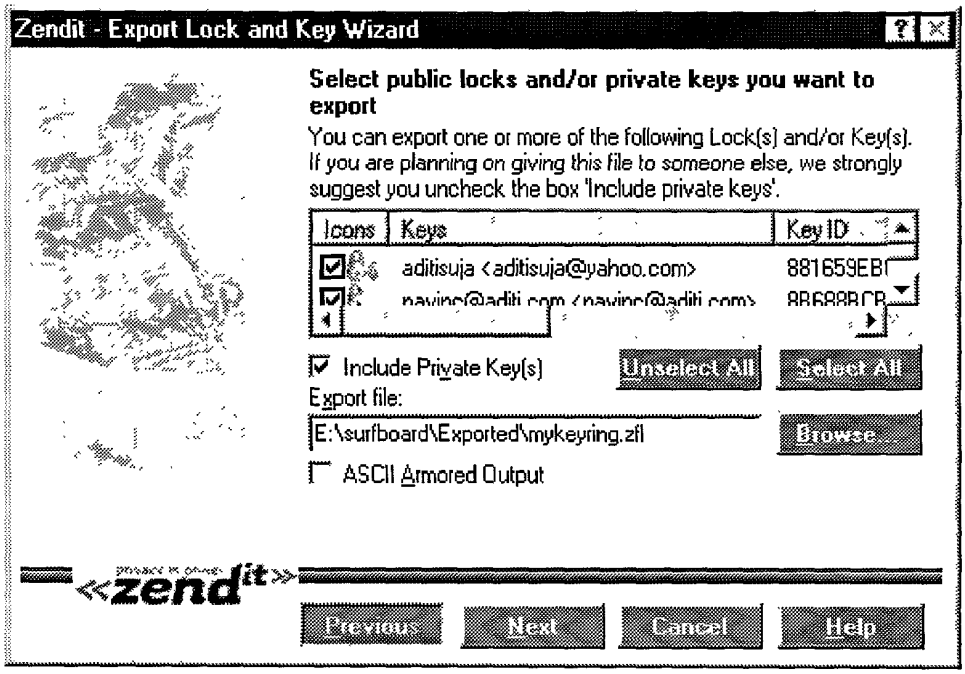


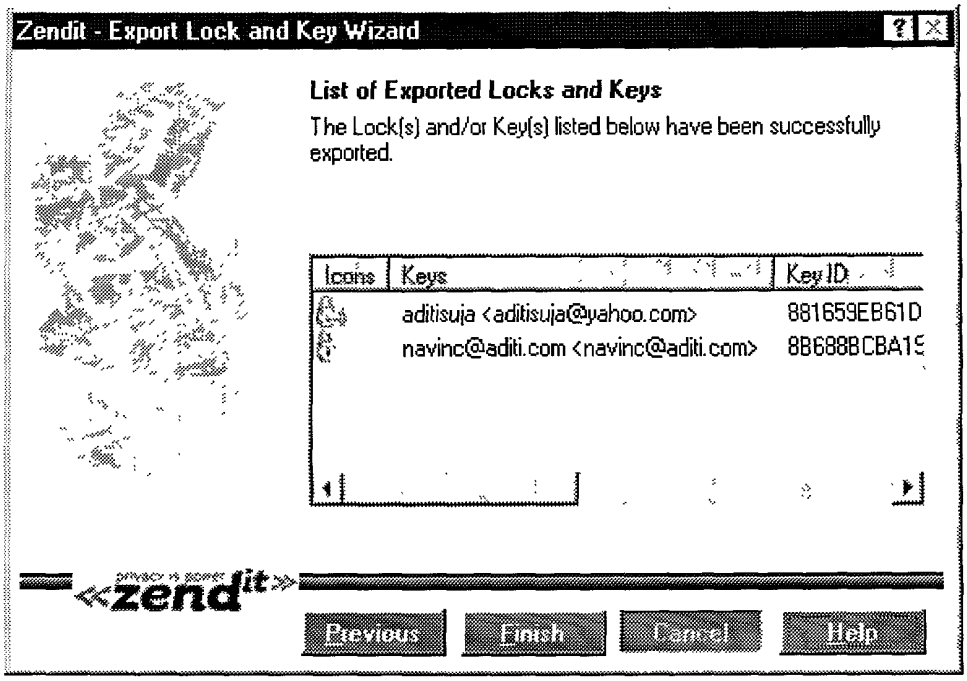


5.12.4 Export a key to a local file

Description

The user selects the key(s) to be exported from the Zendit Vault window. The selected key is exported to the file. The user has an option whether to include the private keys also incase of a lock and key. Also, there is another option to save the file in the Ascii armored form.

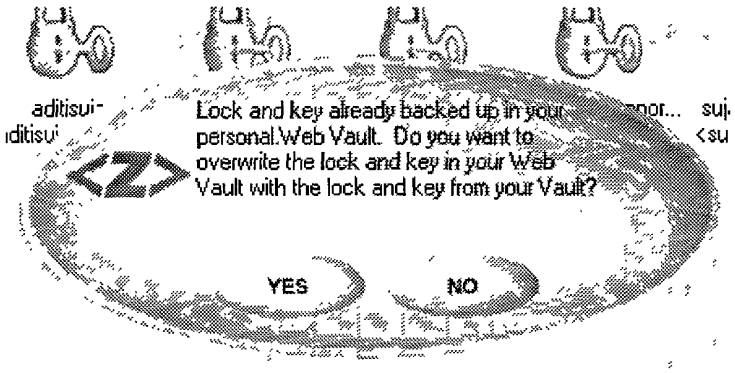




5.12.5 Backup a locks and keys to the web vault

Description

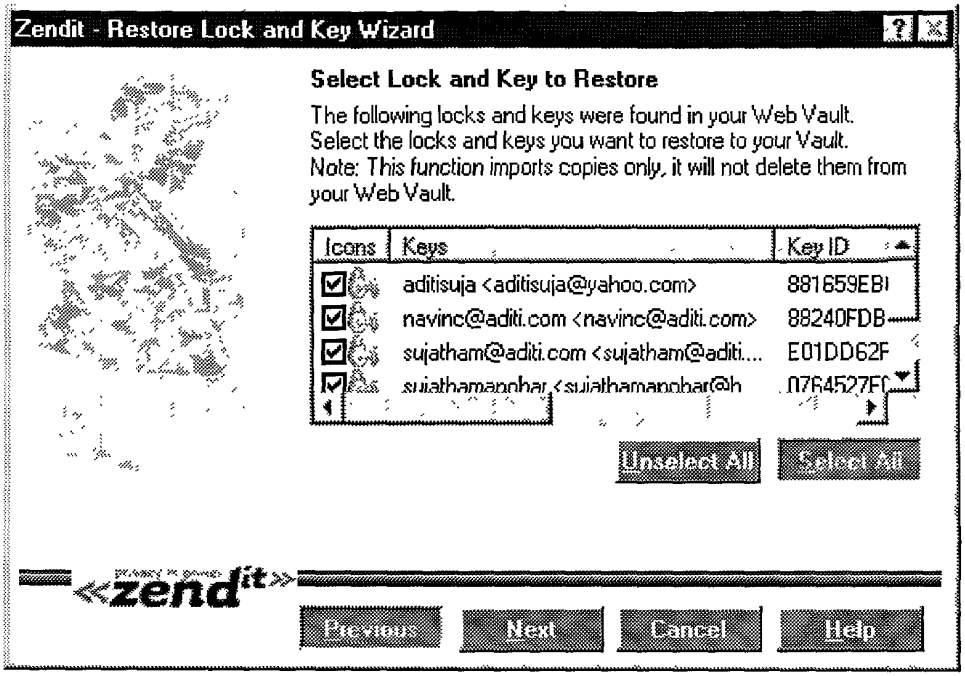
The user can select a lock and key in the vault and backup it in the web vault so that he can restore it back any time.
If the lock and key already exists in the web vault for the same Login ID, he will be given an option to replace it.



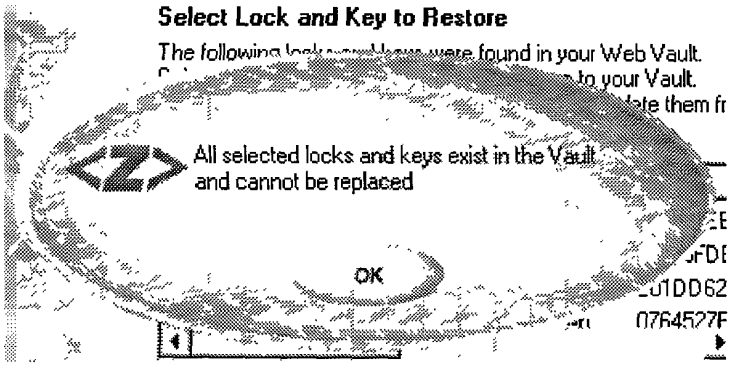
5.12.6 Restore locks and keys to the web vault

Description

When the user selects "Restore" in the Zendit Vault, the following dialog will be brought up where can select the locks and keys to be restored to his Zendit vault. The dialog shows all the locks and the keys the user had backed up to the web vault.



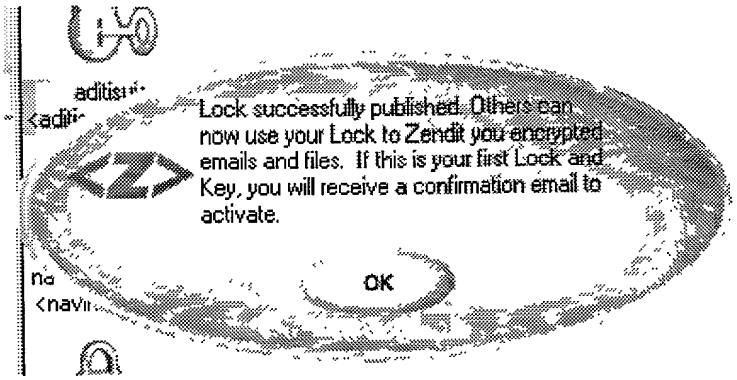
If all the locks and keys that are backed up to the web vault already exists in the zendit vault and if the user chooses "Restore", the following message is shown.



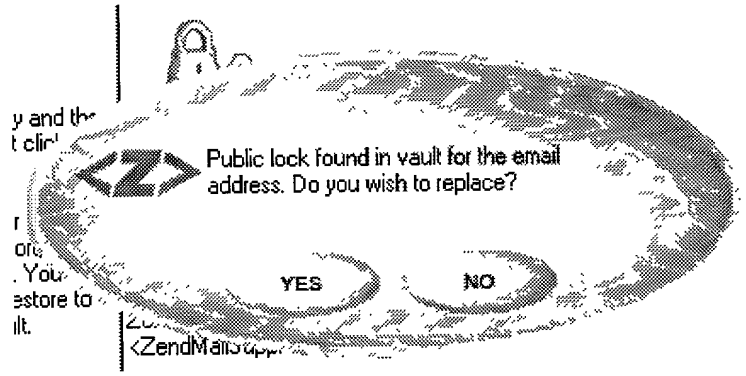
5.12.7 Publish Public Lock

Description

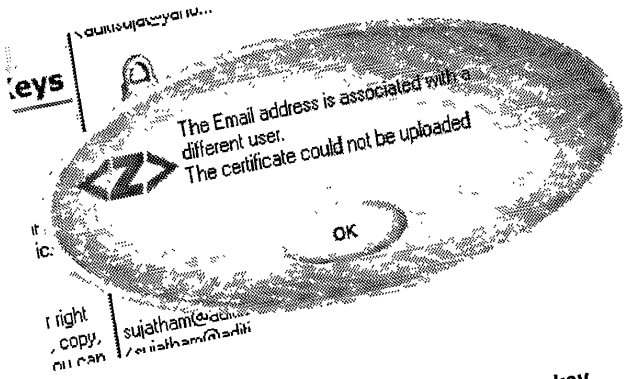
1.When the user chooses a lock and key and selects "Publish Public Lock", the lock is published in the server so that messages can be encrypted for this email address by other Zendit users. If a lock doesnot exist already in the web vault, the following message will be displayed. A confirmation mail is sent to the email address.



2. If a public lock already exists in the web vault for this email address (in tblConfirmedCerts) and is associated for the same Login ID, then the user will be given an option to replace. If the user wish to replace, then the lock which he is trying to publish will be directly inserted into tblConfirmedCerts after deleting the older one. There will not be a confirmation process in this case.



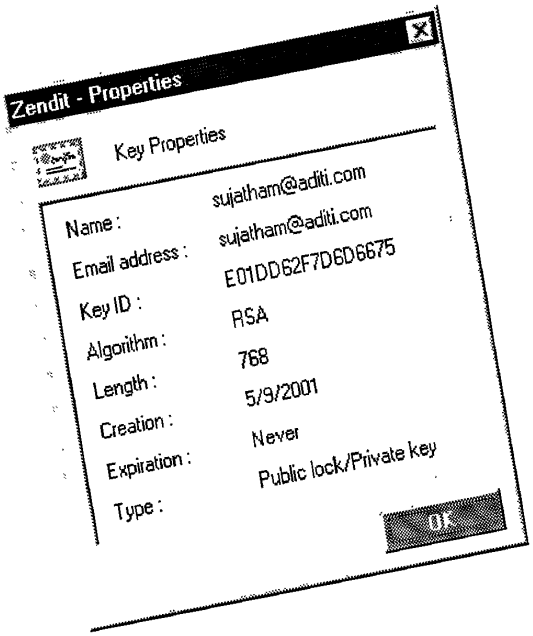
3. If a public lock already exists for this email address in the web vault (either in tblConfirmedCerts or tblConfirmedTempKeys) but associated with different Login ID , the following message will be displayed to the user.



5.12.8 View the certificate associated with the key

Description

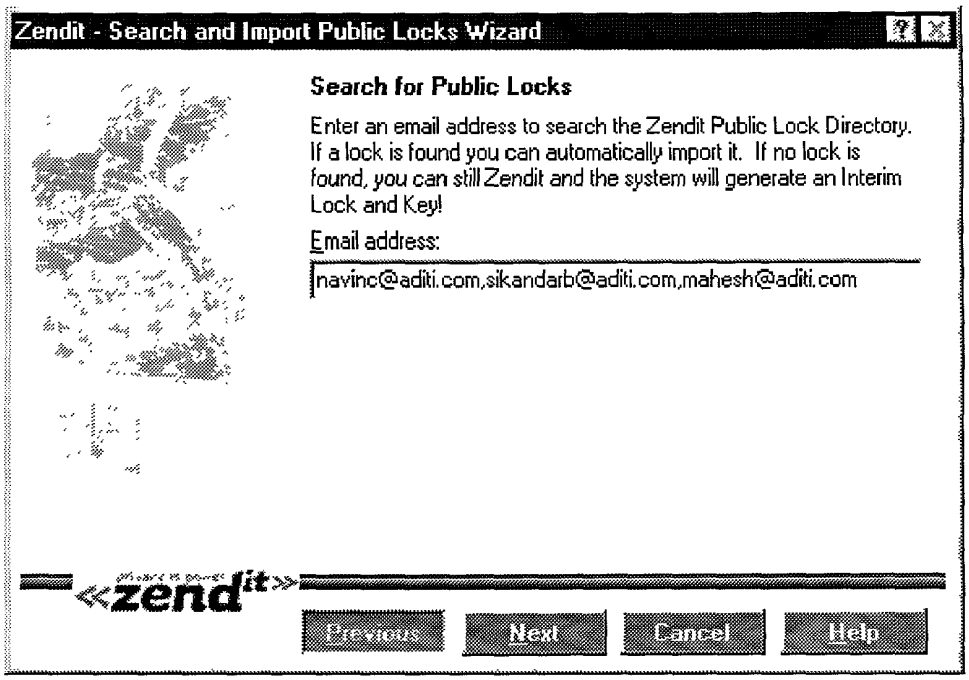
When the user chooses a key to view the certificate, the properties is displayed in the following format.

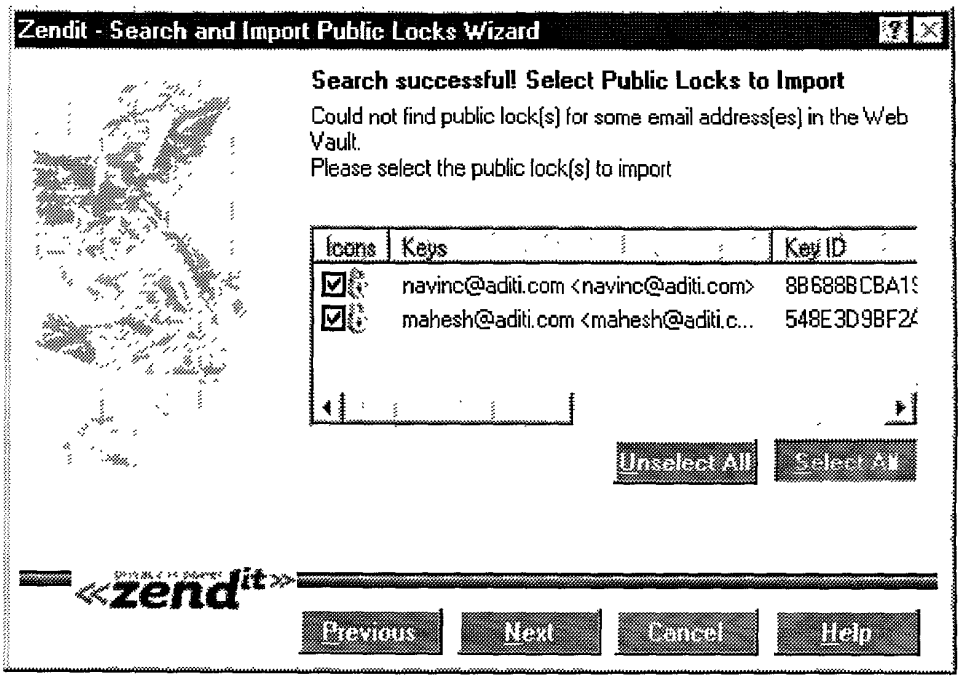


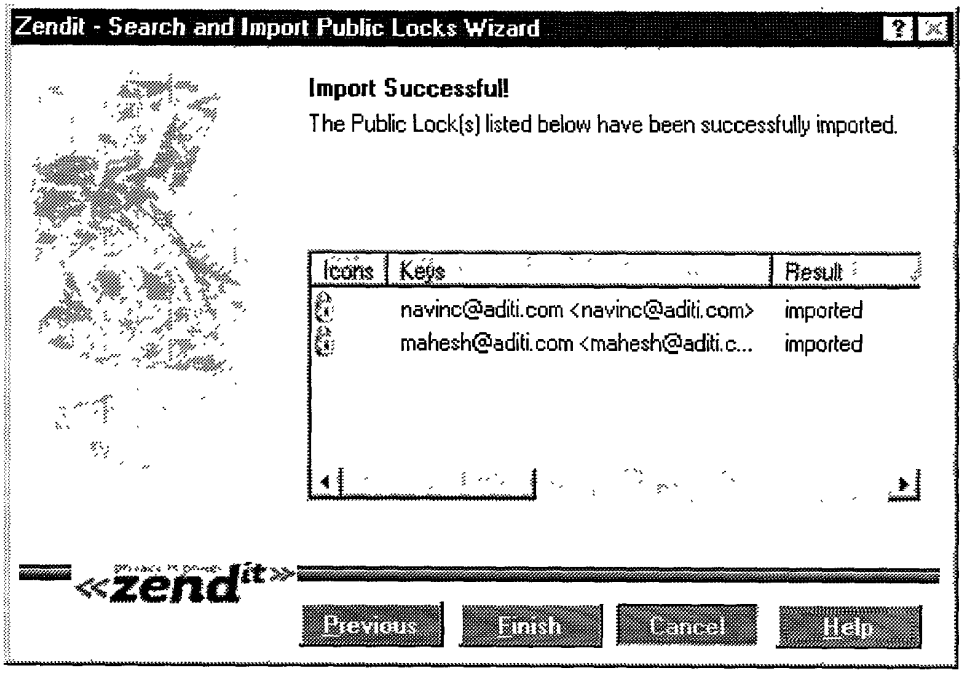
5.12.9 Search for a public lock and import

Description

The user can search for a public lock for a list of email addresses. If the public locks are found in the web vault for at least one of them, the user has an option to import them into his Zendit vault.



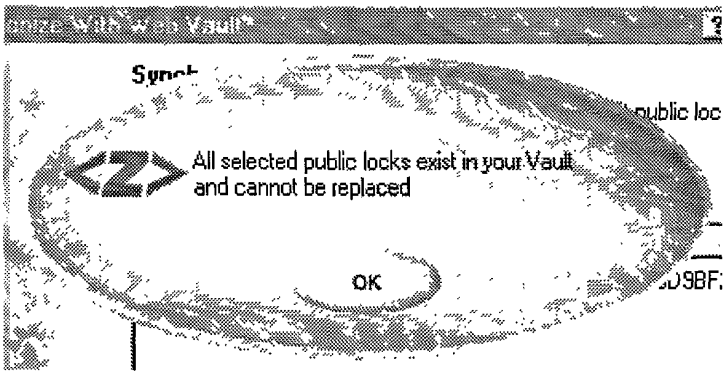

Zendit - Search and Import Public Locks Wizard
Search successful! Select Public Locks to Import
Could not find public lock(s) for some email address(es) in the Web Vault.
Please select the public lock(s) to import
Icons
Keys
Key ID
navinc@aditi.com <navinc@aditi.com>
8B688BCBA1
mahesh@aditi.com <mahesh@aditi.c...
548E3D9BF2
Unselect All
Select All
zendit
Previous
Next
Cancel
Help


Zendit - Search and Import Public Locks Wizard
Import Successful!
The Public Lock(s) listed below have been successfully imported.
Icons
Keys
Result
navinc@aditi.com <navinc@aditi.com>
imported
mahesh@aditi.com <mahesh@aditi.c...
imported
zendit
Previous
Finish
Cancel
Help

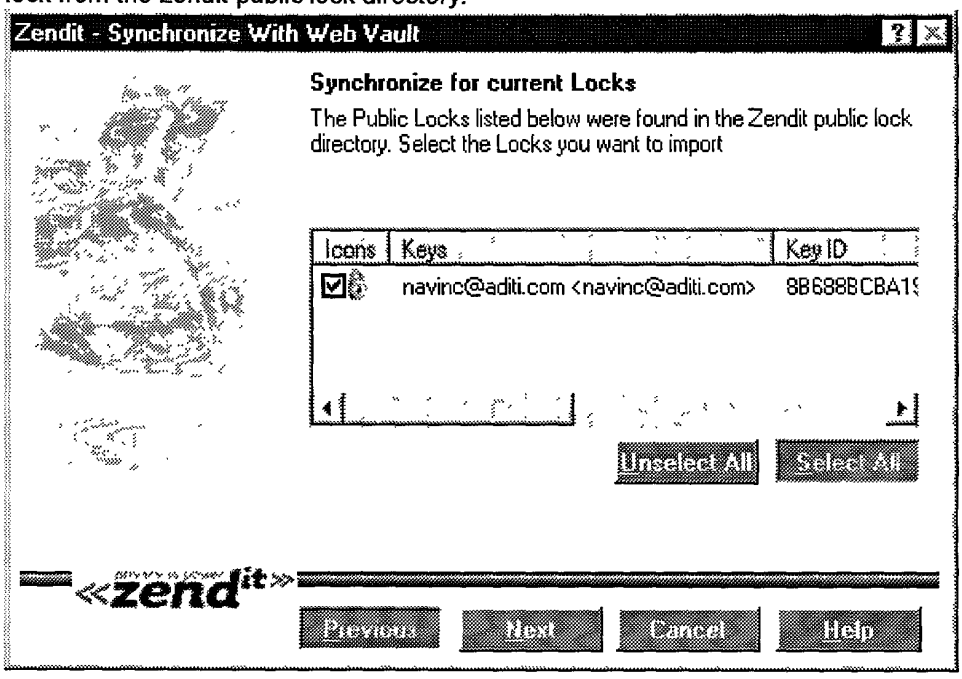
5.12.10 Synchronize with web vault

If a public lock for an email address in the user's zendit vault is different from the one that exists in the zendit public lock directory, the user has an option to synchronize it so that his zendit vault gets updated.

If the public locks for the email address in the zendit vault and in the zendit public lock directory are identical, then the following message is displayed.



If the public locks are different, then the zendit vault brings up the following wizard to import the lock from the zendit public lock directory.



5.13 Key Manager

Description

The basic functionality of this module can be explained in 2 steps.

1. The 'Zendit Vault' is invoked that receives all the inputs given by user though the 'Key management user interface handler'.
2. Specific functions on the 'Cryptographic Service Provider' are called and the appropriate values are passed.

This module also decides the CSP to be used and bridges between the 'Zendit Vault ' and the 'CSP'.

Listed below are the cases in which it takes the input from the Zendit Vault and sends it to CSP via specific function calls.

Functionality	Input from the Vault which goes to CSP via Key Management module	Output from CSP to Key Management module
Create a new keypair	Name Email ID Key pair length Key pair size Date of expiry Pass phrase	KeyID of newly generated keypair
Delete a key /keypair from the key ring.	Key ID Passphrase	Success or Failure indication
Import a key / keypair from a local file into the key ring.	Local file name KeyIDs of keys to be imported	Success or Failure indication
Export a key to a local file.	Key ID	Success or Failure indication
Upload a key to the Zendit server	Key ID	Digital certificate (public lock)
Download a key from the Zendit Server to the key ring	Key ID along with the Digital certificate	Success or Failure indication

View the certificate associated with the key	Key ID	Digital certificate details (such as name, email address, keyID, size and so on).
Generate a revocation certificate for the specified key.	Key ID Revoke key Revoke key size	Not currently used
Edit key passphrase	Key ID Old Passphrase New Passphrase	Success or Failure indication
Change user id on the key	Key ID Passphrase Old User ID New User ID	Success or Failure indication
Query the user's local key rings, a specified file or a specified memory block and provides information about a list of keys / keypairs available in the specified source.	Key ring Key ring size	List of structures, each containing details (such as name, email address, keyID, size and so on) for one digital certificate.
Encrypt file	Cleartext source file name Encrypted destination file name Array of Recipient(s) Number of recipient(s) Array of signer(s) Number of signer(s)	Success or Failure indication
Decrypt file	Encrypted source file name Decrypted destination file name Pass phrase	Flags indicating armored, encrypted and signed status of input content, signature details for each signature found
List Recipients	Encrypted content	List of KeyIDs of keys that can be used to decrypt the encrypted content.

6. Assumptions and dependencies

- OpenLib will be used for encryption and decryption process.
- Zendit will provide the hashing algorithm.
- The SurfBoard would import the certificate only if email ID exists and the certificate is valid.
- The SurfBoard will not support certificate revocation.
- Hashing algorithm is performed both on email id and login id.
- Admin pages will work on IE 4.0 and above.
- The SurfBoard will work only with IE 4.01 and IE 5.01 on Win 98 and Win NT 4.0 Workstation.
- Maximum number of SQL Servers used can be extended to a fixed number of 100 servers.
- When SQL Servers being used are increased, the Zendit system administrators are responsible for moving data across SQL Servers.
- If the Zendit system is not able to send mails, it adds an entry into the error log.
- The system does not support re-sending mails that bounced back.
- Access mechanism for admin pages will be Windows NT challenge-response authentication.
- User can upload more than one public lock but there can be only one public lock associated for an email ID existing on the server.
- Reply to feedback will only be in text format.
- The administrator maintains the correct templates of the various mail systems' web pages.
- Once the user has logged in, the user information is taken from the LSM.
- When the user logs out, the user will be logged out of all the SurfBoard instances and also from the zendit clients i.e outlook and Windows Explorer.
- When the user encrypts (zends) an email, the mail is signed, compressed and encrypted. The encrypted text is then converted into an ASCII armor, which replaces the original message body in the email.
- The OpenLib is viewed as a complete, reliable package and will be used as is. Hence this will not be tested.
- 'SA – FileUp' object from Software Artisans will be used for file upload and download(will be implemented for later version)
- Keys will stored be stored in the key ring as files (as supported by the OpenLib).
- The Zendit system will not be signing the certificates.
- The hardware setup and the required software should exist.
- Support for displaying Advertisements will not be provided in Version 1.0 of the system.

7. Error Handling and Messages

This section explains the methodology that is to be followed for error handling and error messages for the Zendit web site and the SurfBoard.

7.1 Web Site

The standard for handling errors throughout the web site would be to indicate server-related errors as "Problem at the Server. Please try again or send a mail to support@zendit.com "

The details of the server-related errors would be logged in the Event Log. This event log has to be manually monitored and addressed by one of the Zendit administrators. The user interface to view this log would be using the NT Event Viewer.

Validation error messages that are thrown up to the user would be displayed on the same page itself. For easy maintenance all the messages would be contained in an include file.

7.2 Surfboard

Validations would be performed in every relevant function. These would be detailed in the low level design.

Some of the specific exceptions identified during the low level design would be trapped using exception handlers, and user-friendly error messages would be displayed.

The description of the error messages would be maintained in a Resource file.

Templates Document

1.0 Overview

This document details the usage of different types of templates. It aims to provide the ZendIt administrator with the insight of how the template entries can be derived.

2.0 Template Types

There 3 types of templates:

- Send Templates
- Click Templates
- Read Templates

The Send and Click templates are used during the 'Zend' operation. The Read template is used during the 'DeZend' operation

Note: All the templates are retrieved by using the ZendView Source utility and not by using the explorer's view source functionality.

3.0 Zend Operation

The process followed by the Surfboard while 'Zending' a mail is detailed in this section. The section under Send Templates helps the administrator with retrieving the 'Send' templates. The click templates are discussed with respect to sending the mail under the section 'Click' templates.

3.1 Send Templates

The send templates are used to identify the template to be used while trying to the zend a mail. These templates are also used for retrieving the values of the 'To', 'Cc', 'Bcc' and 'Body' fields from a compose page of a mail system. The activities performed by the surfboard while 'zending' the mail are detailed as follows.

1. Get the URL of the main document from the Address bar.
   For e.g., this URL would be
   http://lw4fd.law4.hotmail.msn.com/cgi-bin/compose?a=96813929223044116693 4
   which is the URL for the compose page in Hotmail.

2. Check for the mail system in the URL
   The surfboard obtains the list of templates available. Each template has a 'Mail system' field. For e.g., for the hotmail site, the mail system value would be represented as 'Hotmail'.

3. The surfboard goes through the template list and checks for the existence of the 'Mail system' value in the domain part of the URL.

   For e.g., in the case of hotmail, the 'mail system' value 'hotmail' is checked for in the 'http:// lw4fd.law4.**hotmail**.msn.com/' string. (i.e., it tries to find the mail system string within the first '/' excluding the 'http://' string.

4. If a template matches with the mail system, then the surfboard checks if the current page has frames or not and accordingly builds a internal list of matching templates from the available templates. The surfboard makes use of the 'Frame type' field in the templates to get the list of the frame based, non-frame based and iframe-based templates.

5. The surfboard then looks at the 'URL Phrase' value to determine if the user is on a valid page from which a mail can be 'Zent'.

   For e.g., in the Hotmail system, the template that was built for 'Compose' page has the URL Phrase as '/cgi-bin/compose?'. This string is checked in the main URL that the Surfboard obtains.

6. If the system is non-frame based, but the document has an iFrame imbedded in it, then the template is entered as an iFrame based template. The surfboard will look for the different fields from which the values for 'To', 'Cc', 'Bcc' etc is obtained.

7. If the system is non-frame based and the document has no iFrames in it, then the template is entered as non-frame based template. The surfboard will look for the different fields from which the values for 'To', 'Cc', 'Bcc' etc can be obtained.

8. If the system is frame-based, the corresponding templates are entered as frame based templates. The surfboard will go through each of the frames in the document and tries to match the 'Frame URL' field value in the document URL of each frame. Once a match is found, the Surfboard uses this template to retrieve the 'To', 'Cc', 'Bcc' etc values.

9. Once the matching template is found, the message in the 'Body' is encrypted with the public locks of all email addresses found in 'To', 'Cc' and 'Bcc' fields and placed back in the 'Body' text area. All these elements are retrieved using the 'Form name' entry. The 'To', 'Cc', 'Bcc' and the 'Body' elements are part of the form specified in the 'Form name' entry.

10. The Surfboard, then zends the encrypted mail by simulating the send functionality of the mail system using the click templates.

3.2 Click Templates

The auto-send of the encrypted mail is accomplished by using the click templates. A send template has an associated set of click templates. These click templates are used to determine the element in the document, which on clicking, simulates the mail system's send operation.

A click template consists of the relational ID, element-type, attribute-name and attribute value fields. Each element has certain attributes and values. A set of click templates identifies an element that is clicked by surfboard to simulate the send option. Thus the element is described by element type, attribute name and attribute value combinations. The relational ID of a set of click templates is used to identify single element. A different set of click templates with a different relational ID denotes a different element. A send template can have different sets of click templates by which the Surfboard can simulate the send option.

For example, if in a compose page of a mail system, the message can be sent by clicking the following elements,

<input type="submit" name="Send.x" value="Send" id="Send" onclick="onSubmitCompose(1);">

And

<img src = "/images/iSend.gif" border = 0>

The click templates associated with the send template will have the following entries.

Relational ID	Element Type	Attribute Name	Attribute Value
1	INPUT	type	submit
1	INPUT	name	Send.x
1	INPUT	value	Send
2	IMG	src	/images/iSend.gif
2	IMG	border	0

This means send option can be simulated by clicking the element INPUT with attributes type, name and value which has values submit, Send.x and Send respectively OR by clicking the element IMG with the attributes src and border with values /images/iSend.gif and 0 respectively.

4.0 DZend operation

The 'Read' template is made use of during the Dzend operation. The template is used to fetch the sender's email ID. This is to verify the signature of the sender in the encrypted message. The decryption process does not stop if the sender's email ID is not found. The process continues without the decrypted message being verified. The method by which the read templates are derived is detailed in the subsequent section.

4.1 Read Templates

The string between 'Header Start' and 'Header End' field values is retrieved. In the string retrieved, a sub-string between 'Sender Start' and 'Sender End' field values is obtained. This string would give the 'Sender's information. But to obtain the email ID from this string, the 'Token Begin' and 'Token End' field values are used.

For e.g., in the Yahoo mail read page, the source contains the following.

....
...
<INPUT type=submit value=Move name=MOV>
</FONT></TD></TR></FORM></TBODY></TABLE>
<TABLE cellSpacing=0 cellPadding=0 width="100%">
<TBODY>
<TR vAlign=top>
<TD>
<TABLE cellSpacing=0 cellPadding=1>
<TBODY>
<TR>
<TD vAlign=top noWrap align=right><B>From:</B></TD>
<TD>"center" <center@hotmail.com><SMALL> | <B><A
href="/ym/BlockSender?NBR=1&NE=center@hotmail.com&box=Inbox&MsgId=42
01_3181411_22684_879_1762_0&inc=&num=&Search=&YY=75044&or
der=down&sort=date&pos=0">Block address</A></B></SMALL></TD></TR>
<TR>

<TD vAlign=top noWrap align=right><B>To:</B></TD>
<TD><navcha@yahoo.com></TD></TR>
<TR>
<TD vAlign=top noWrap align=right><B>Subject:</B></TD>
<TD></TD></TR>
<TR>
<TD vAlign=top noWrap align=right><B>Date:</B></TD>
<TD>Sat, 2 Sep 2000 07:50:29 +0530</TD></TR></TBODY></TABLE></TD>
<TD vAlign=top>
<FORM
action=http://address.mail.yahoo.com/yab/us?v=YM&cmode=1&Lang=us&nf=1
method=post>.....
....

The Header Start the read template can be specified as <INPUT type=submit value=Move name=MOV>
The Header End can be specified as <TD vAlign=top noWrap align=right><B>Date

The string retrieved between these two values is

</FONT></TD></TR></FORM></TBODY></TABLE>
<TABLE cellSpacing=0 cellPadding=0 width="100%">
<TBODY>
<TR vAlign=top>
<TD>
<TABLE cellSpacing=0 cellPadding=1>
<TBODY>
<TR>
<TD vAlign=top noWrap align=right><B>From:</B></TD>
<TD>"center" <center@hotmail.com><SMALL> | <B><A
href="/ym/BlockSender?NBR=1&NE=center@hotmail.com&box=Inbox&MsgId=42
01_3181411_22684_879_1762_0&inc=&num=&Search=&YY=75044&or
der=down&sort=date&pos=0">Block address</A></B></SMALL></TD></TR>
<TR>
<TD vAlign=top noWrap align=right><B>To:</B></TD>
<TD><navcha@yahoo.com></TD></TR>
<TR>
<TD vAlign=top noWrap align=right><B>Subject:</B></TD>
<TD></TD></TR>
<TR>

From this string, the surfboard fetches the string between the 'Sender Start' and 'Sender End' field values. The Sender Start can be specified as <TD vAlign=top noWrap align=right><B>From:
The Sender End can be specified as <SMALL> |

The string retrieved between these two values is
</B></TD><TD>"center" <center@hotmail.com>

From this string, the surfboard fetches the string between the 'Token Begin' and 'Token End' field values. The Token Begin can be specified as <
The Token End can be specified as >

The string that is retrieved between these two values would give the sender's email ID (center@hotmail.com).

5.0 Template Blob

The send, click and read templates are stored in separate tables. Changing any values of these templates does not reflect in the Surfboard. As the number of web based mail system that ZendIt support increases, we will end up having huge number of templates. For example, Yahoo mail system has around 10 templates. If we support 30 mail systems, the number of templates will around 300. Each template can be around 150 bytes, which means user has to download 45KB of data. Since very large template data is to be downloaded to the client, the template structure understandable by the Surfboard is generated, compressed, symmetrically encrypted and stored in a separate table. Clicking on the 'Create Blob' button in the admin templates page performs this activity. The application variable that stores the current template version gets updated in the respective system only after the template blob is created. The application variables have to be refreshed in the all other web servers for the changes to be reflected across the application. The new templates will be downloaded the next time user logs in to the system. When the surfboard makes a request for templates, the ZendIt webserver will send only the Template Blob and not the templates that are stored in a different table.

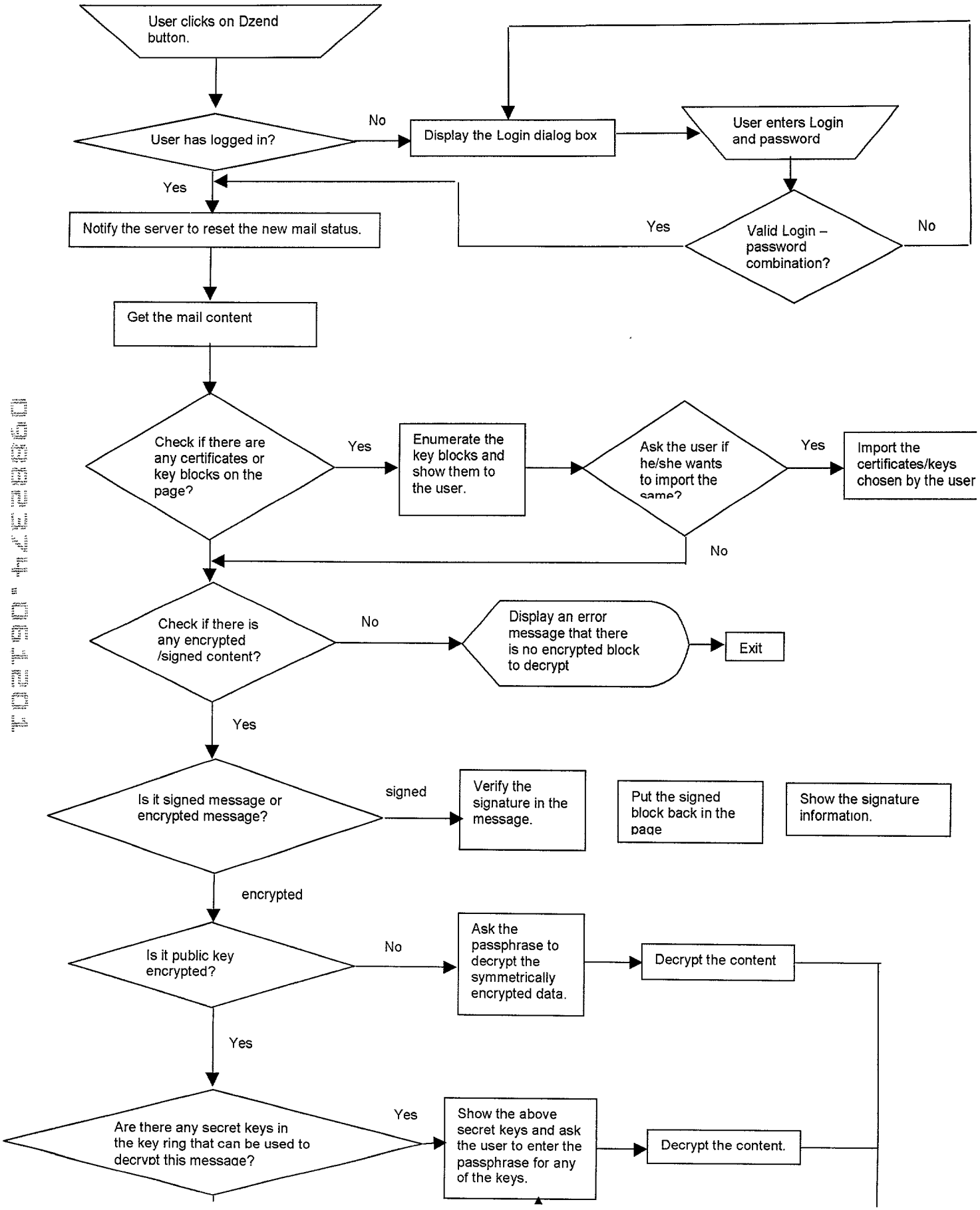
Note: It is very important that when a template is added or modified, you should update the template blob so that the subsequent template request from the client (Surfboard) will get the new template.

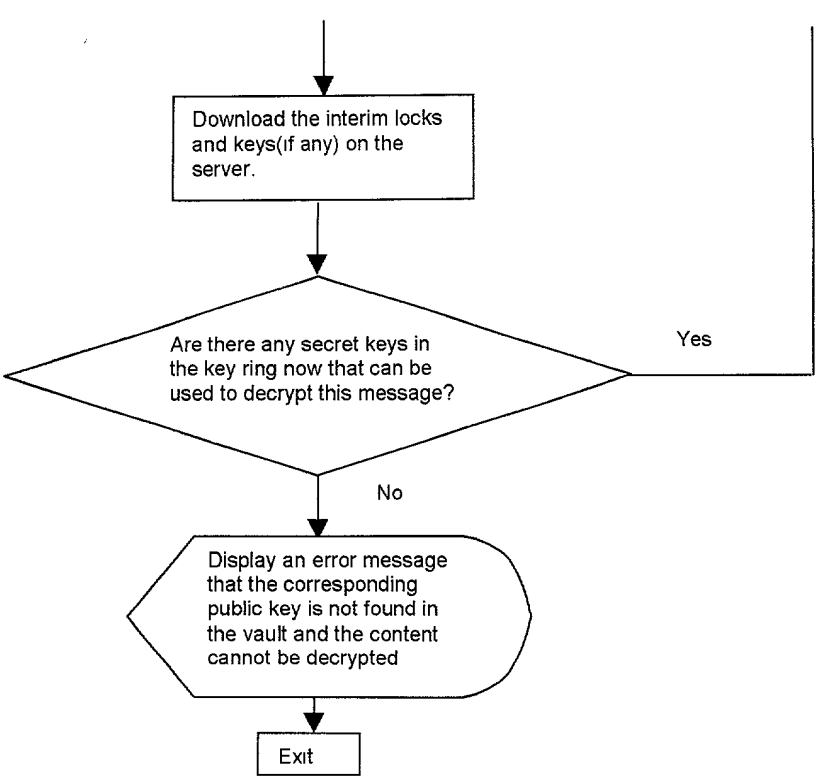
The 'Create Blob' functionality needs the SymmEncComp.dll component to be installed on the web servers. Those template changes made after creation of template blob will not be reflected in the Surfboard.

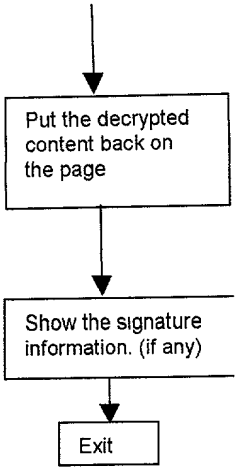
6.0 Conclusion

The process by which the different types of templates are derived and used by the Zend and DZend process has been explained. This is to help the ZendIt administrator to retrieve the templates and also to be a reference for templates usage.

DZend Flow in IE / outlook:

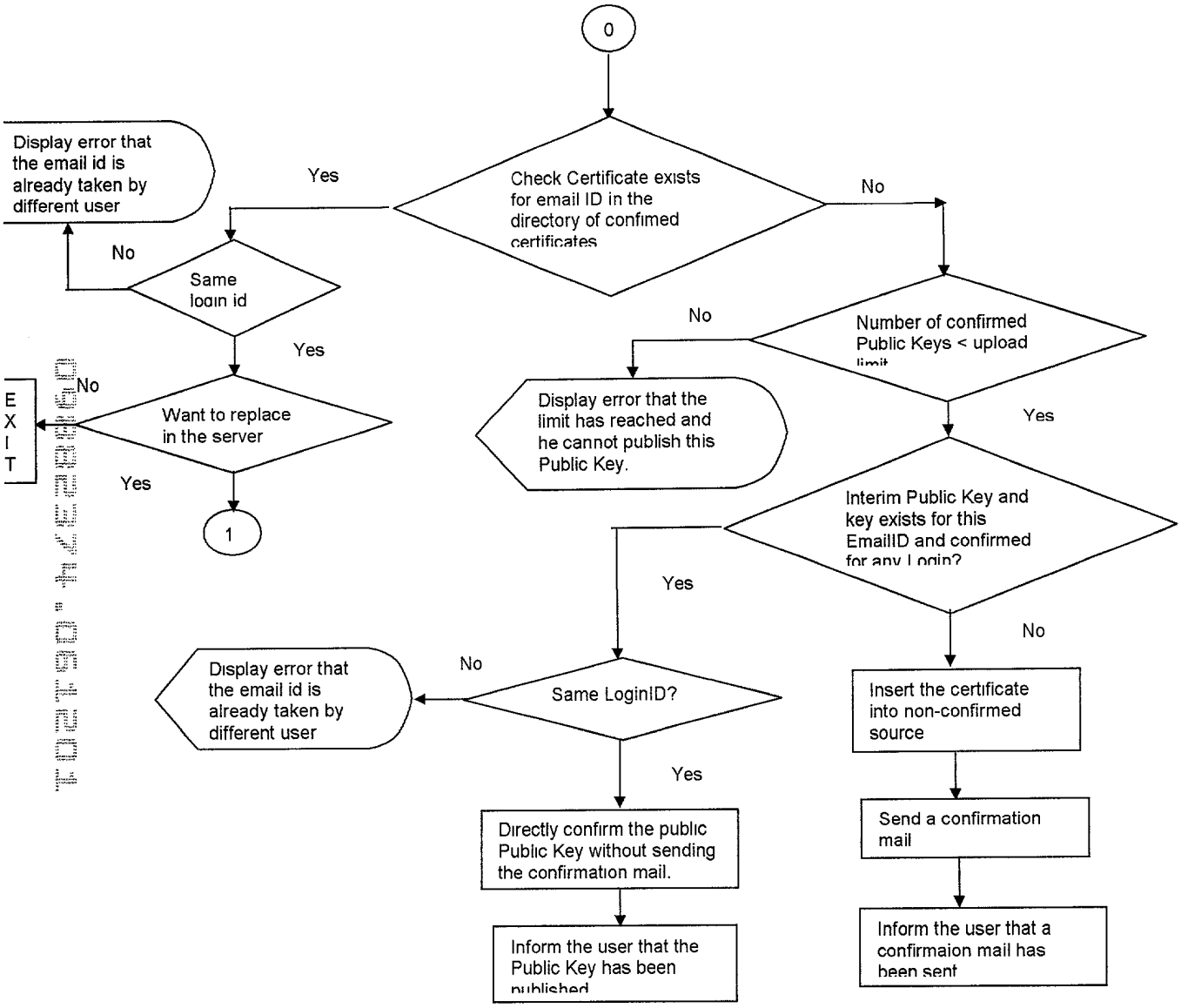


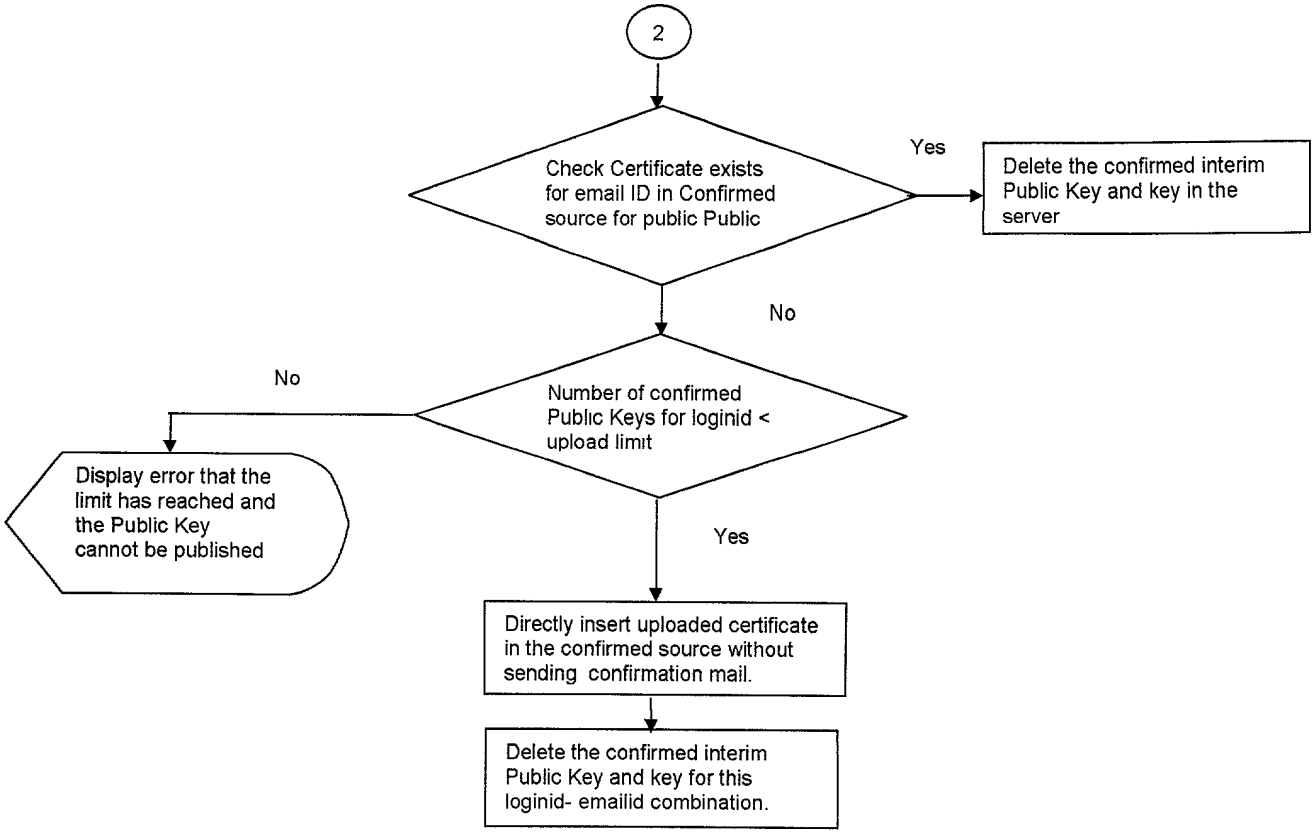




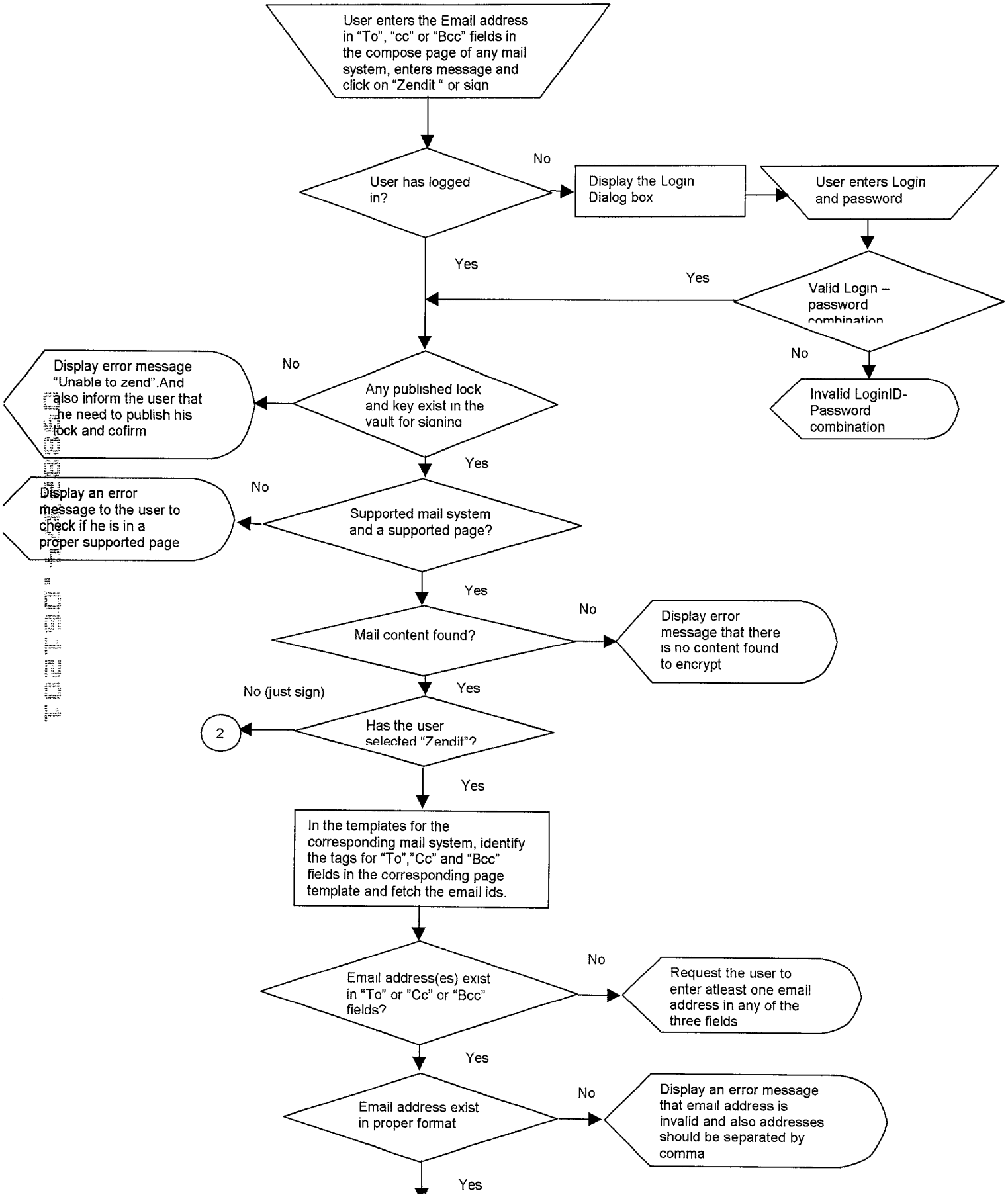
Publish Public Public Key Process Flow

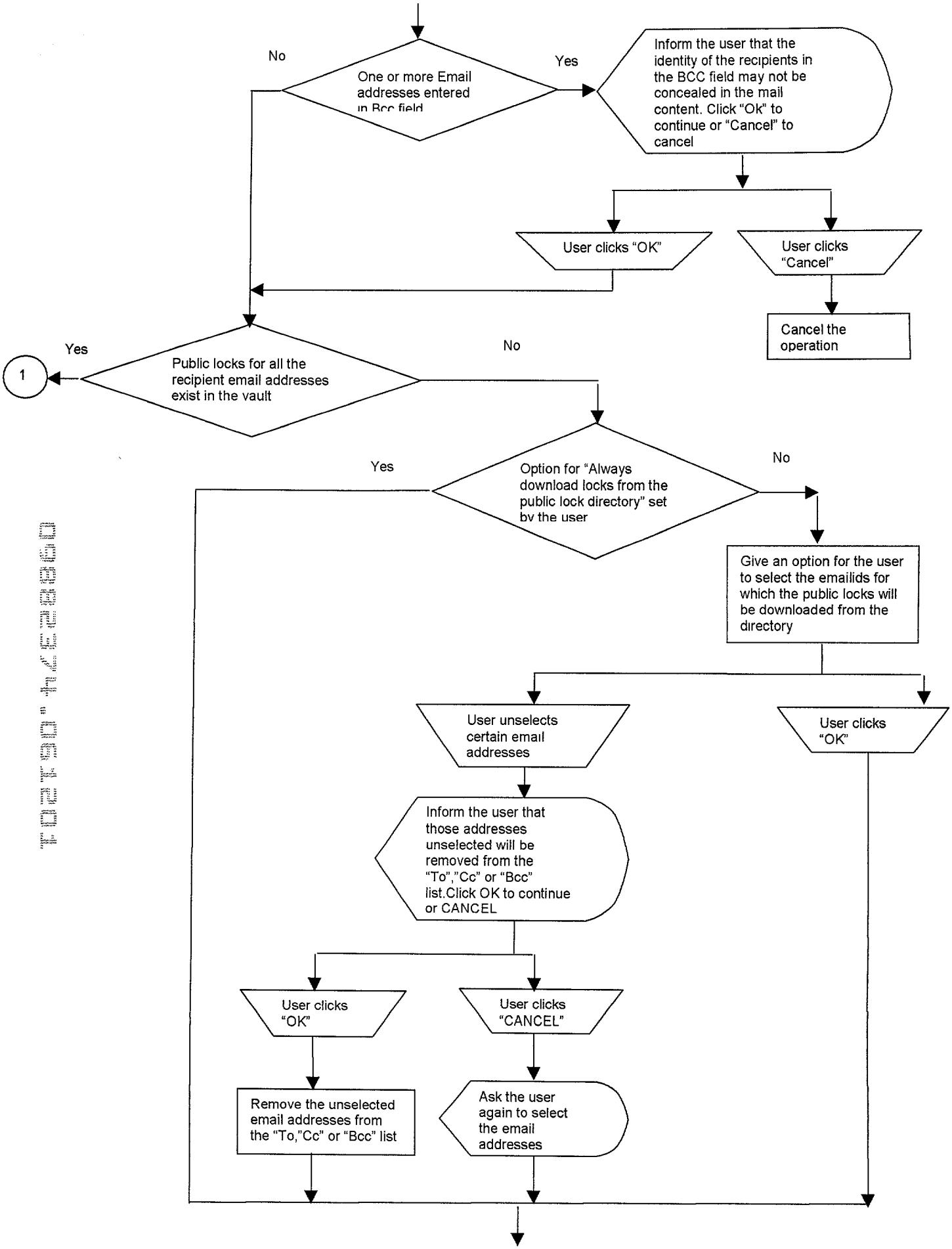
Case 0: Publish a normal public Public Key
Case 1: Replace the existing Public Key in the server
Case 2: Publish a downloaded interim Public Key and key (automatically happens when the user downloads the interim Public Key and key from the server)

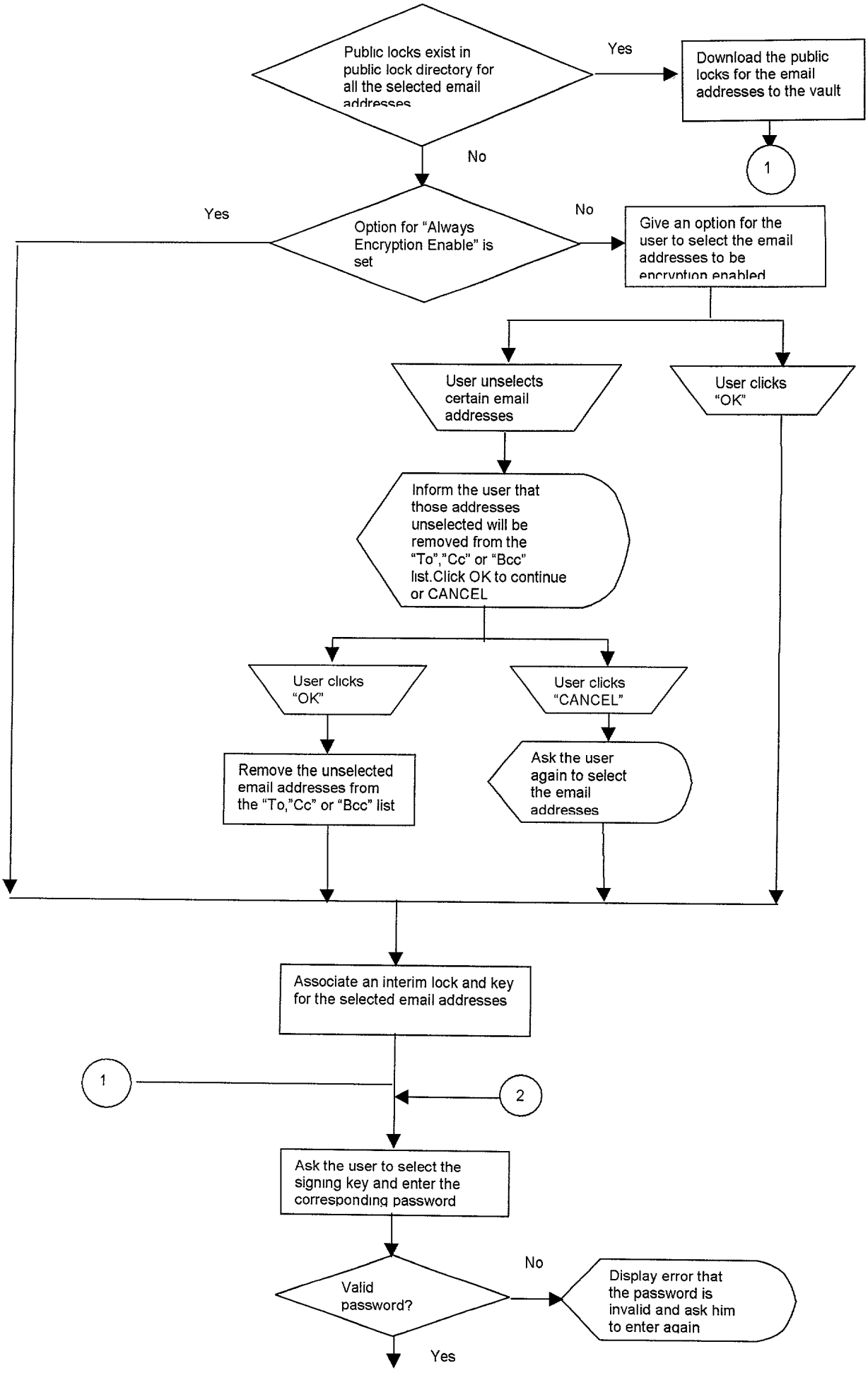


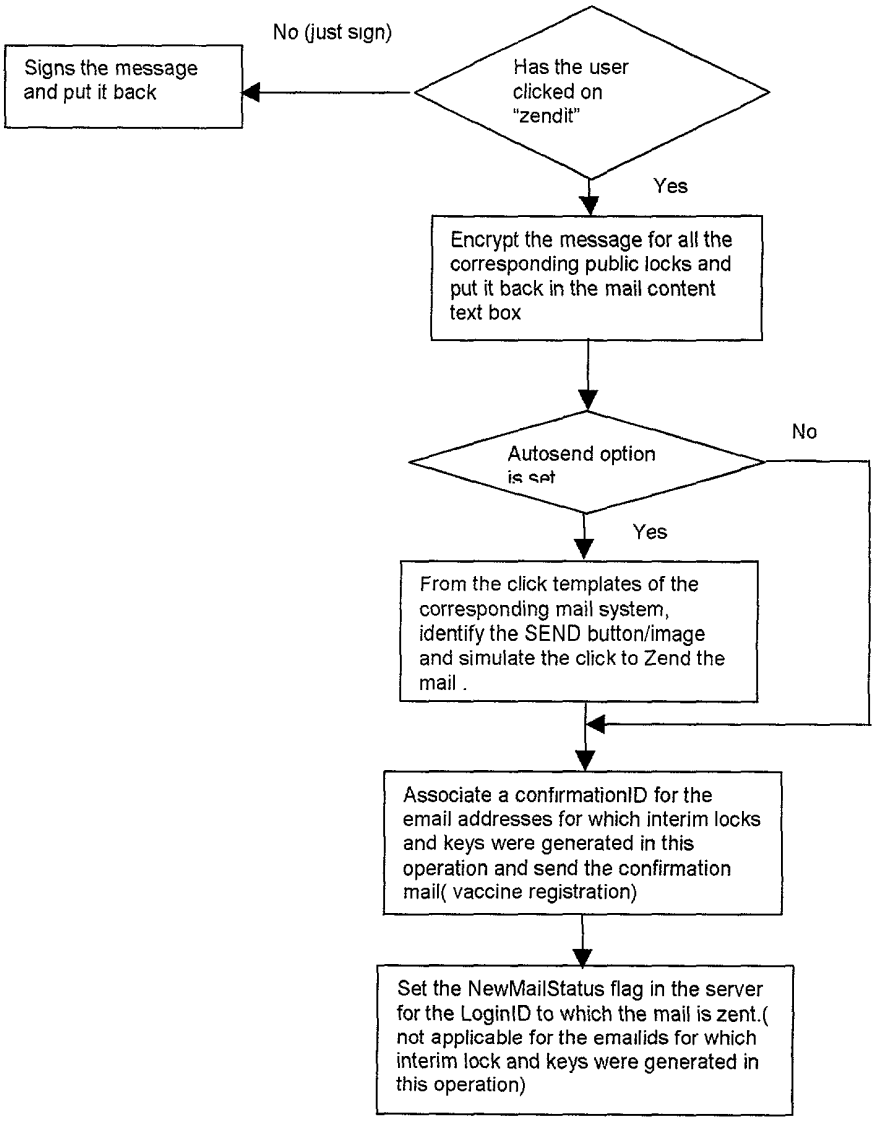


Zend process in IE

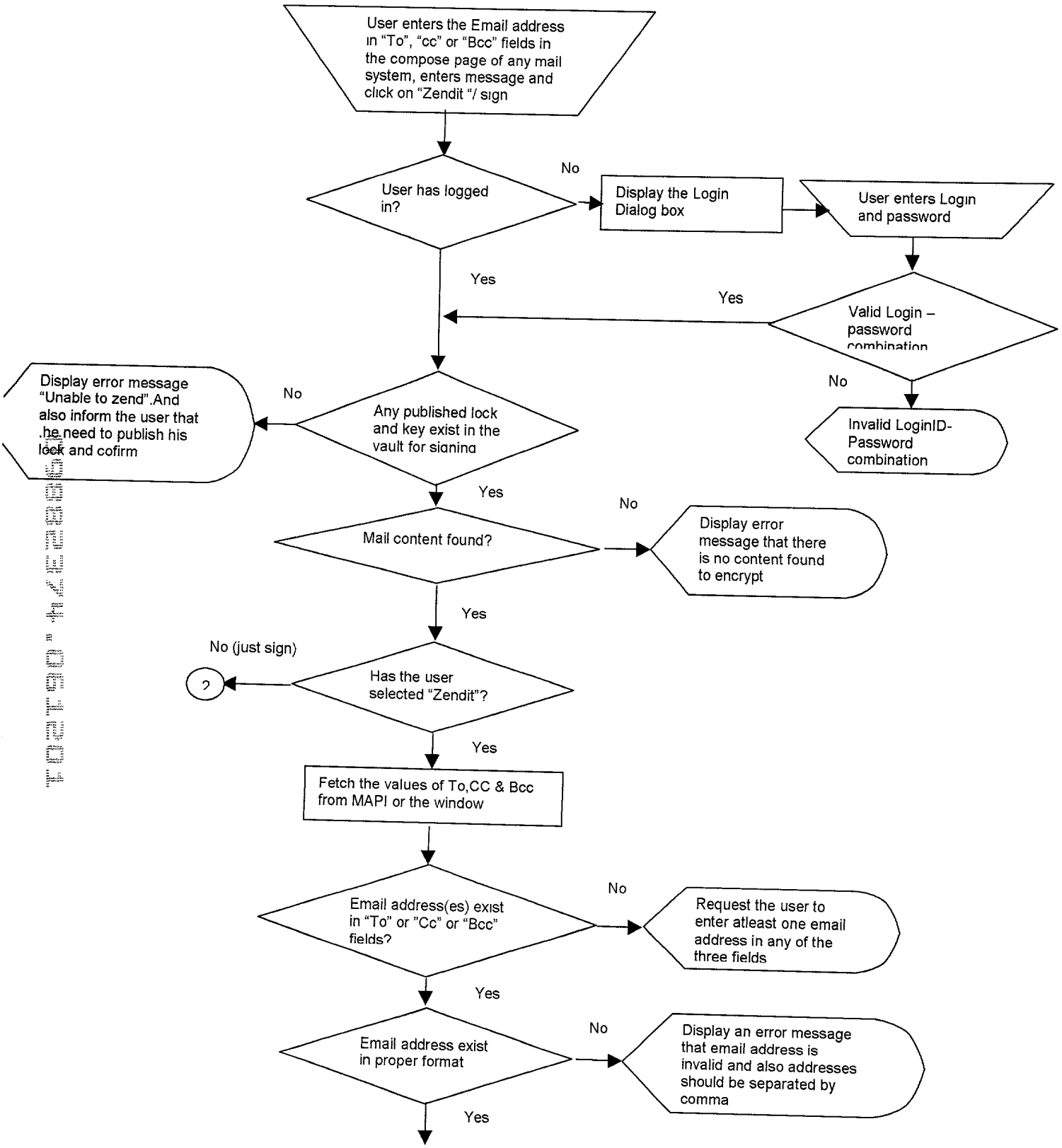


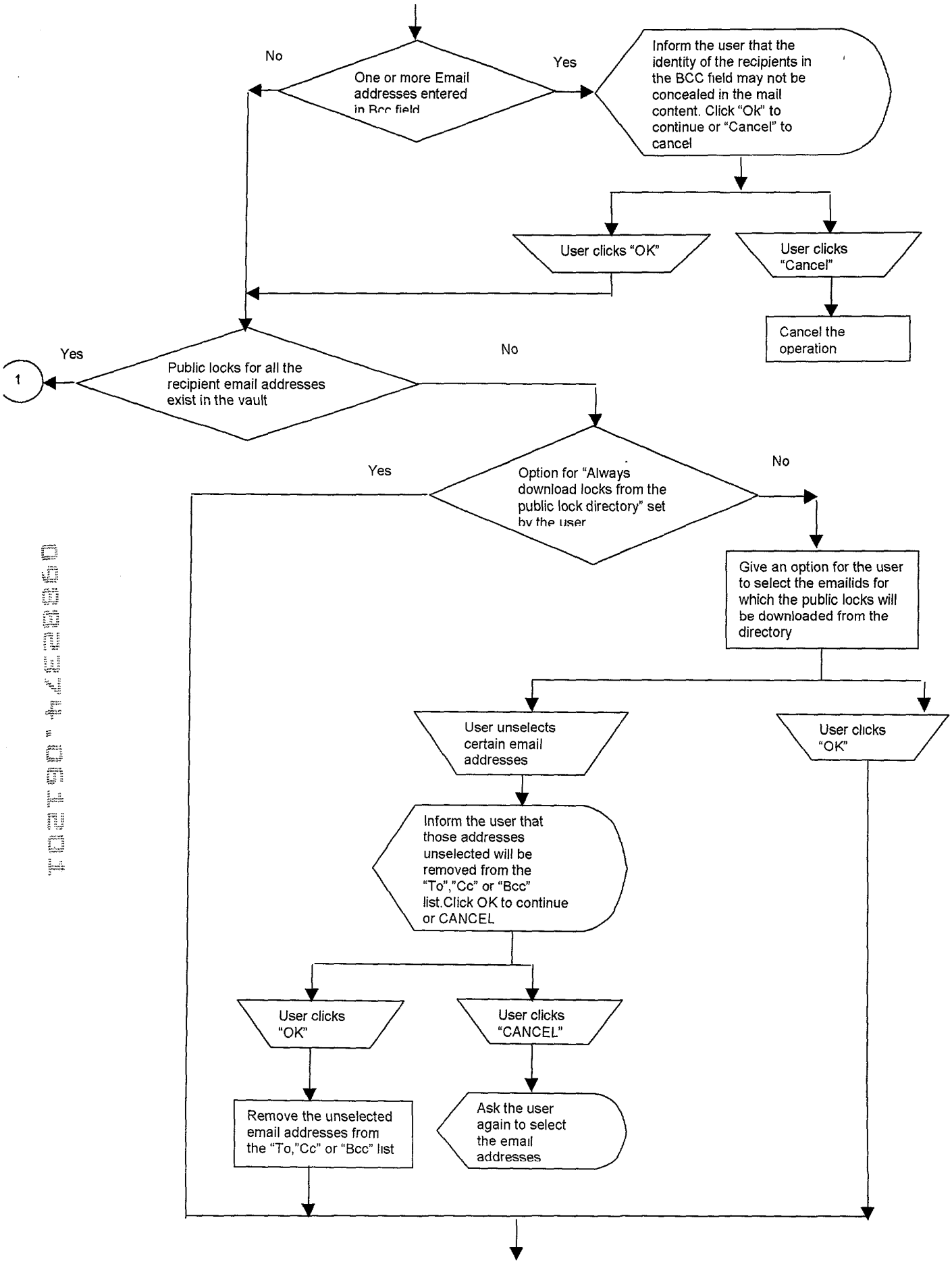


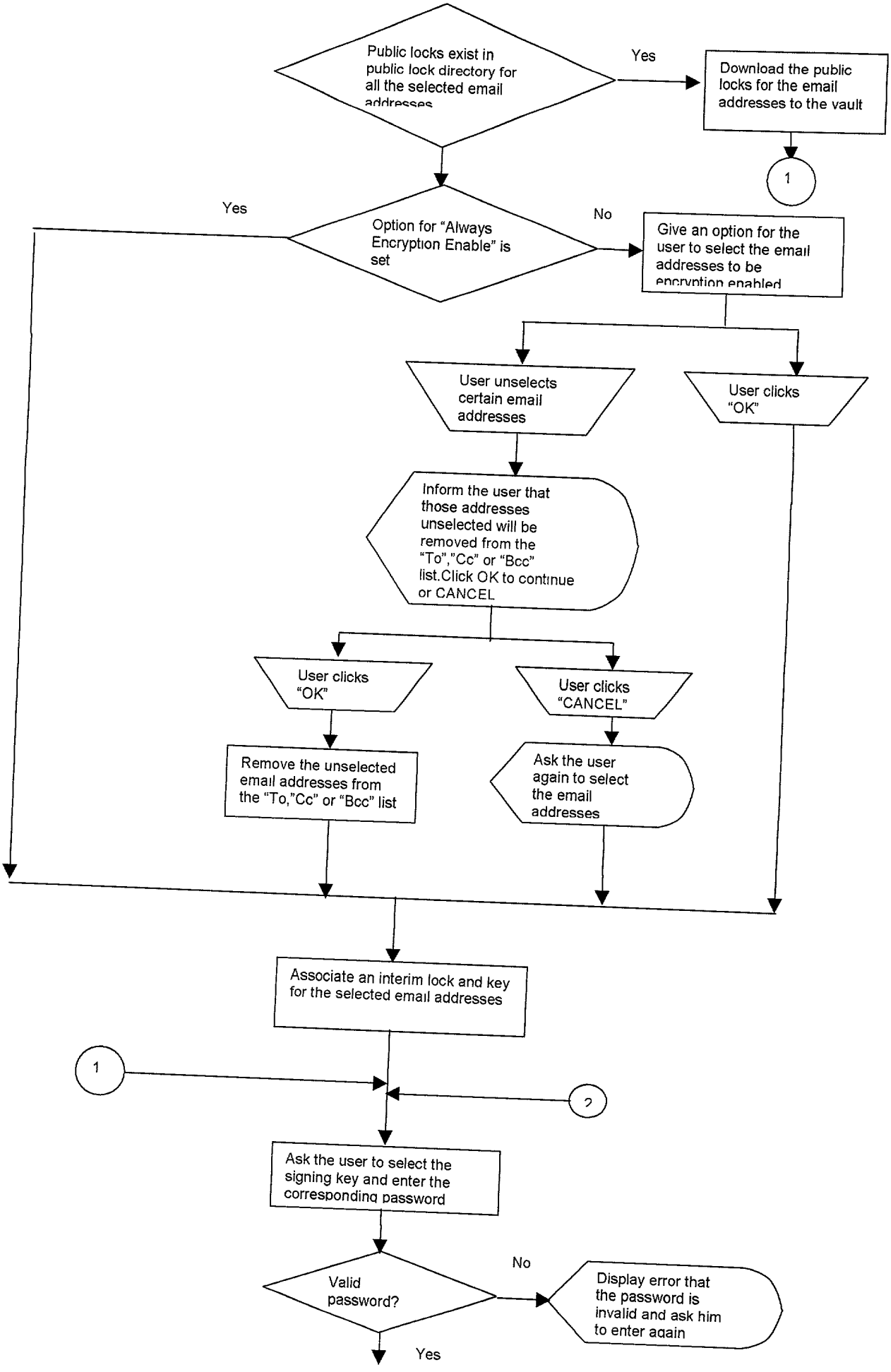


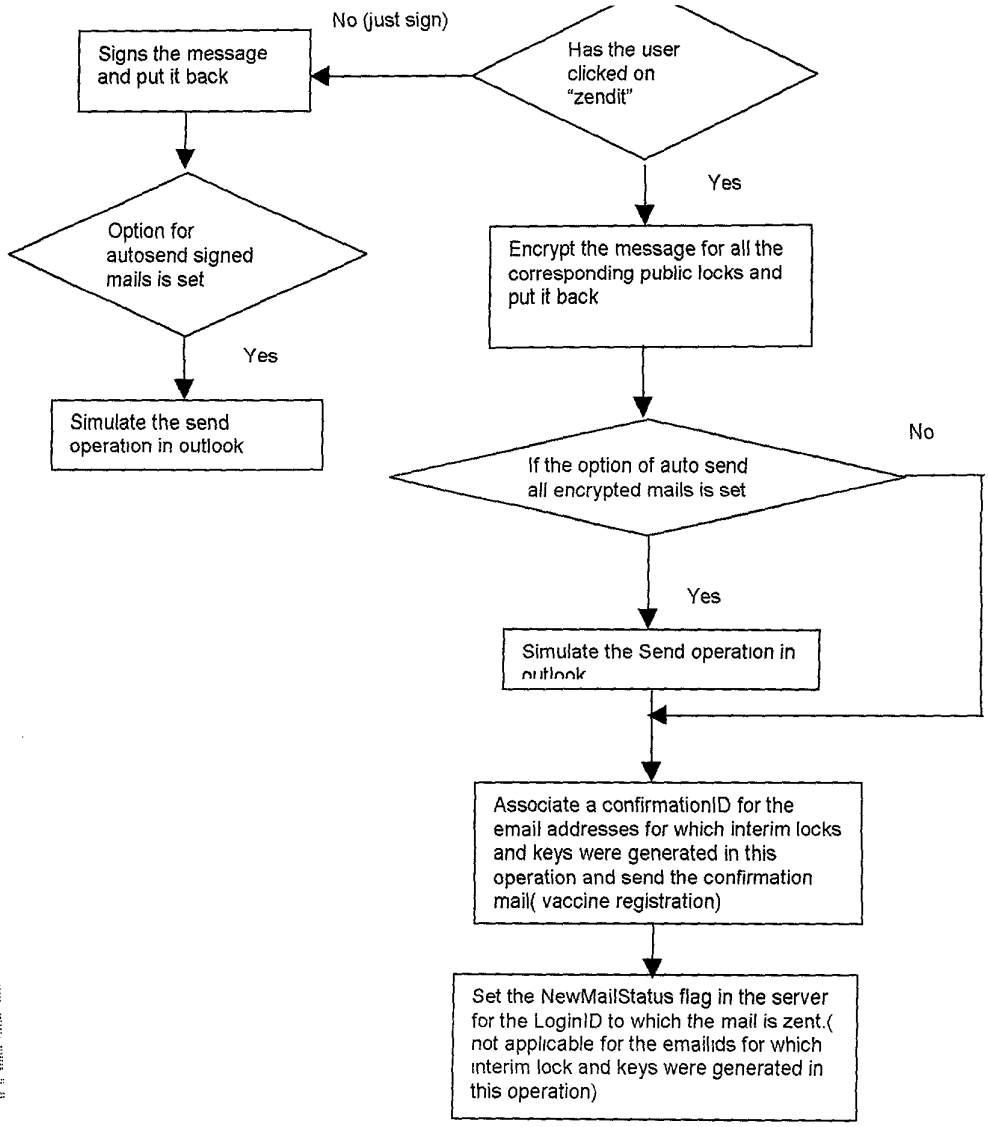


Zend Process in Outlook:






Public locks exist in public lock directory for all the selected email addresses
Yes
Download the public locks for the email addresses to the vault
1
Yes
Option for "Always Encryption Enable" is set
No
Give an option for the user to select the email addresses to be encryption enabled
User unselects certain email addresses
User clicks "OK"
Inform the user that those addresses unselected will be removed from the "To","Cc" or "Bcc" list.Click OK to continue or CANCEL
User clicks "OK"
User clicks "CANCEL"
Remove the unselected email addresses from the "To,"Cc" or "Bcc" list
Ask the user again to select the email addresses
Associate an interim lock and key for the selected email addresses
1
2
Ask the user to select the signing key and enter the corresponding password
Valid password?
No
Display error that the password is invalid and ask him to enter again
Yes



Quick Registration

Back Search Favorites History

http://www.zendit.com/default.asp

Go

privacy is power «zendit»

Join Now

Join & Download Zendit

The Art of Encryption

(n-kript)

Providing World Class e-confidentiality solutions.

try quick registration!

QUICK REGISTRATION

A fast convenient registration process. Quick Registration encryption-enables your email address FAST...with a Lock and Key generated by Zendit. We absolutely delete your private key after you import! You create new Lock and Keys at any time. 1, email confirmation vs. 2!

NOID REGISTRATION

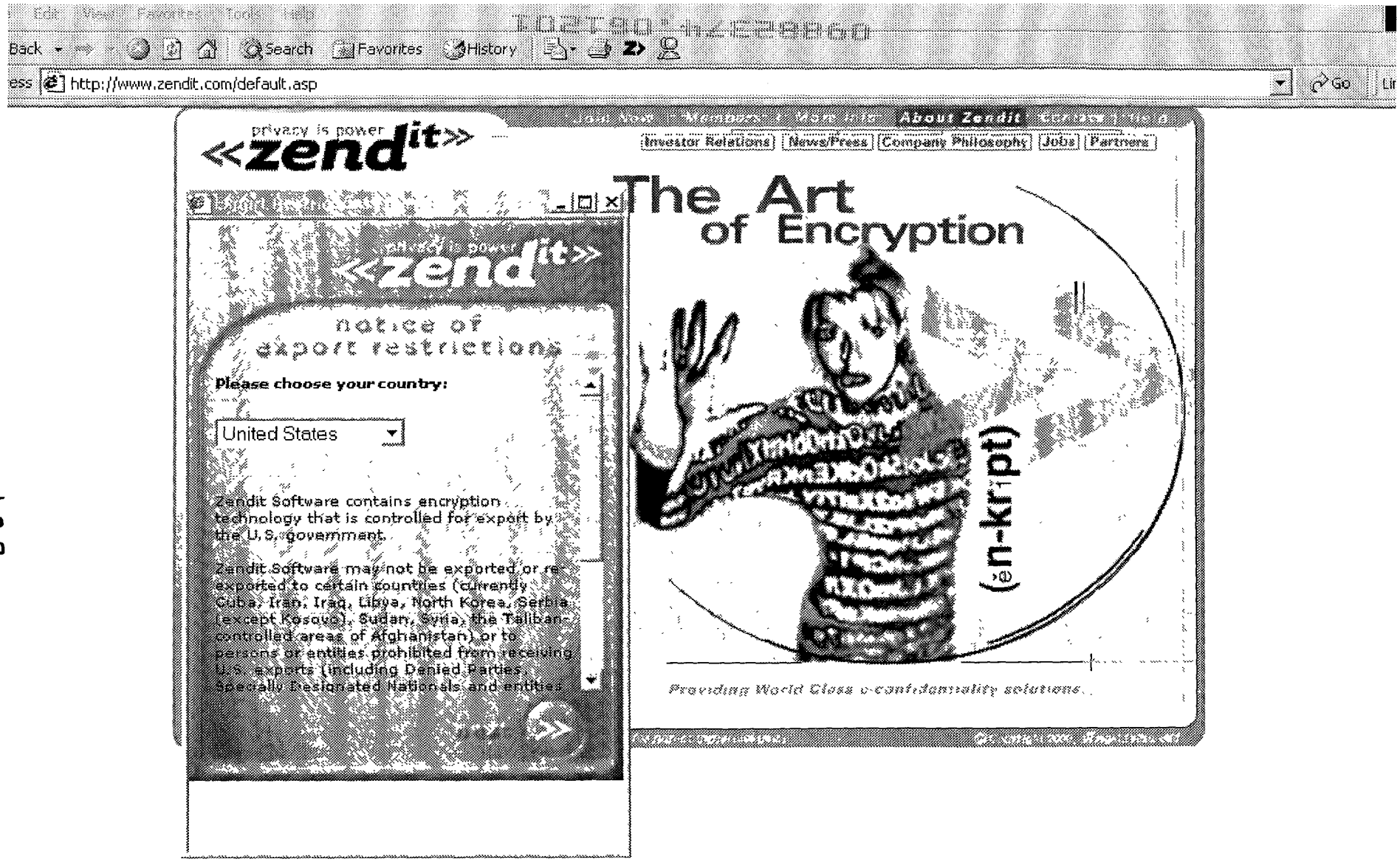
Or, use our traditional registration also known as Noid registration. This registration includes time consuming steps for additional security including 2 email confirmations vs. 1. You generate your Lock and Key pair.

(best for the crypto literate)

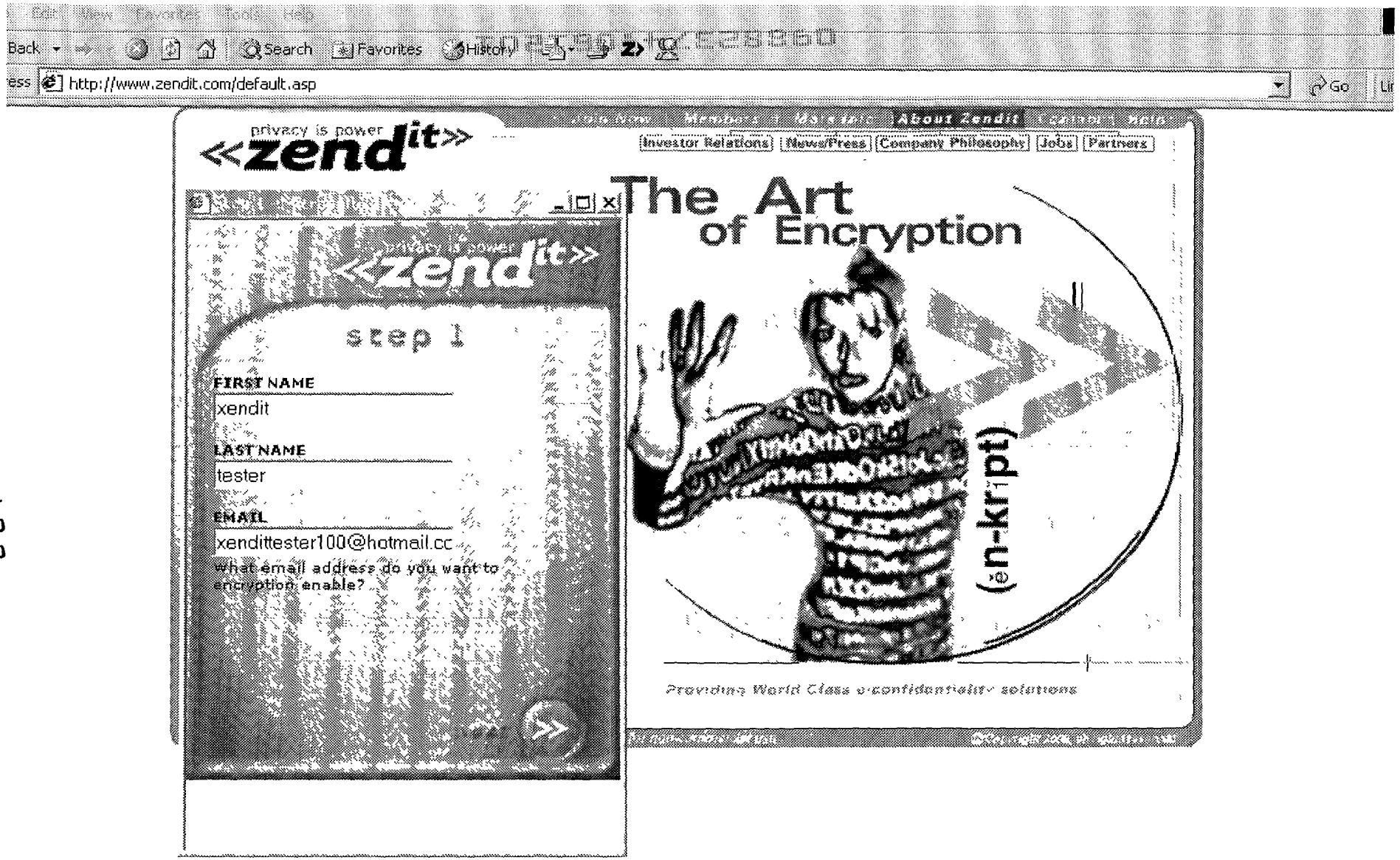
Login Zend DZend ZPanel Vault Help

Please login to activate the Surfboard

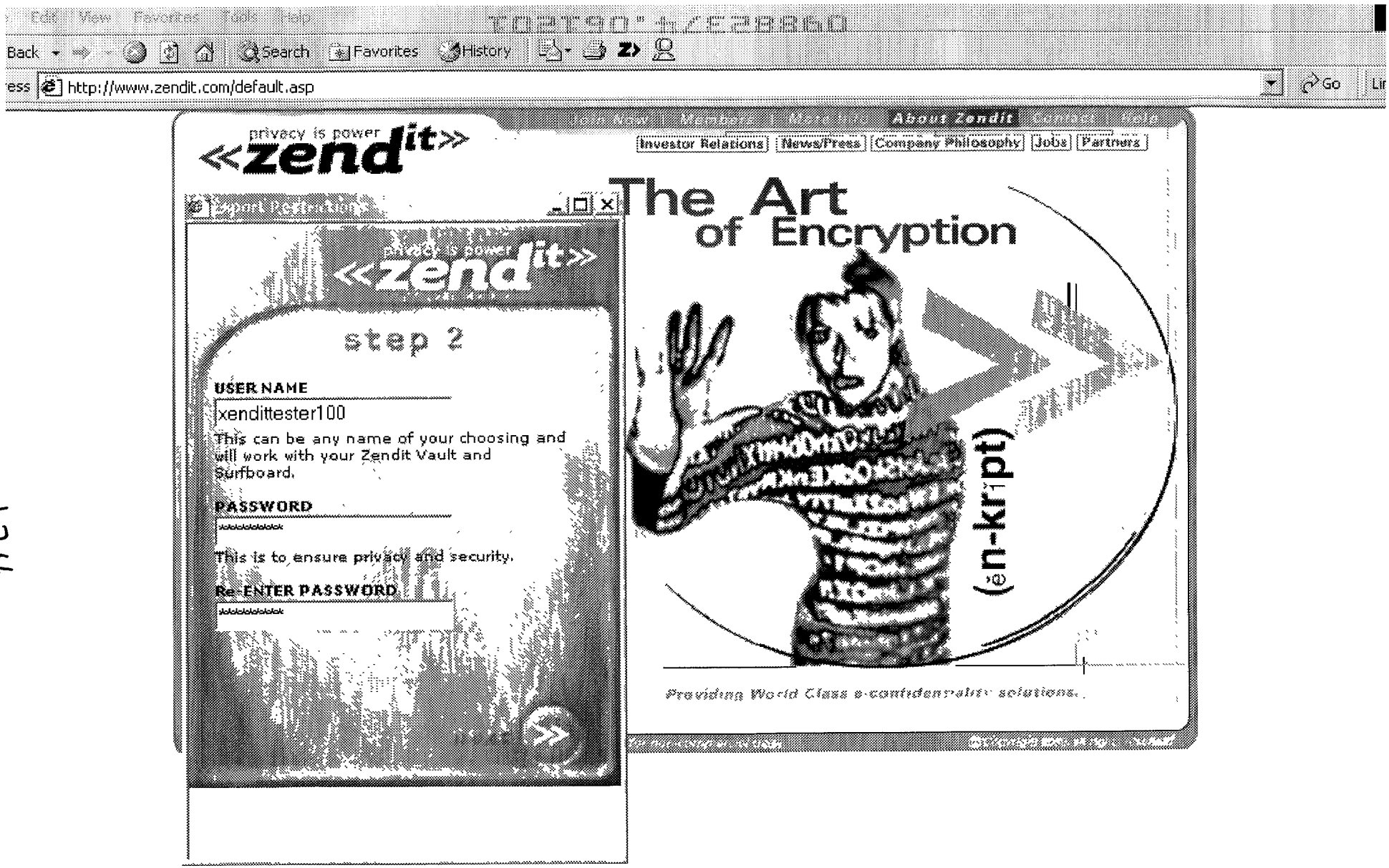
Internet

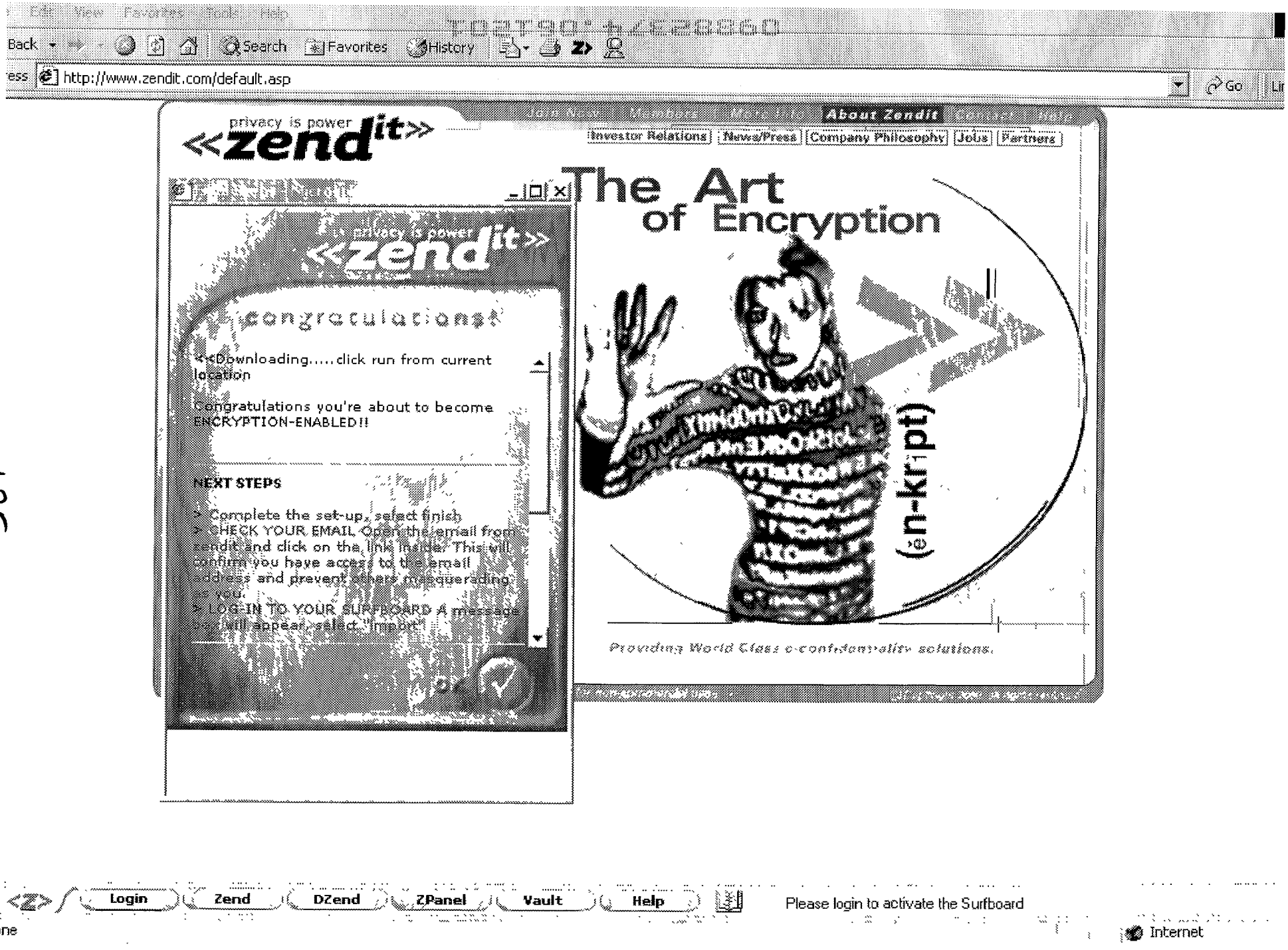

Back
Search
Favorites
History
http://www.zendit.com/default.asp
privacy is power
zendit
About Zendit
Investor Relations
News/Press
Company Philosophy
Jobs
Partners
notice of export restrictions
Please choose your country:
United States
Zendit Software contains encryption technology that is controlled for export by the U.S. government.
Zendit Software may not be exported or re-exported to certain countries (currently Cuba, Iran, Iraq, Libya, North Korea, Serbia (except Kosovo), Sudan, Syria, the Taliban-controlled areas of Afghanistan) or to persons or entities prohibited from receiving U.S. exports (including Denied Parties, Specially Designated Nationals and entities
The Art of Encryption
(en-kript)
Providing World Class e-confidentiality solutions
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Internet

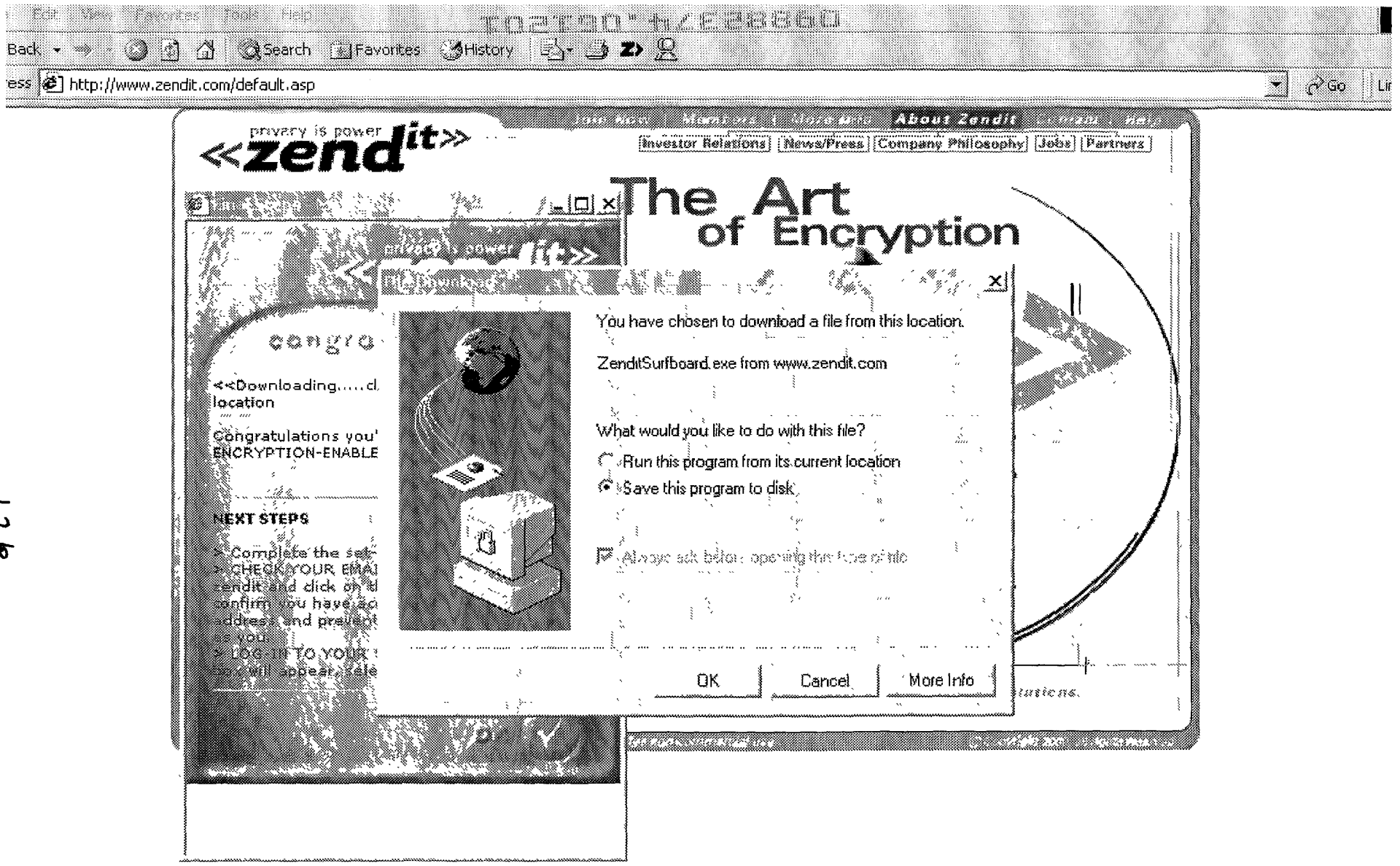
122


Back
Search
Favorites
History
http://www.zendit.com/default.asp
Go
privacy is power
zendit
Investor Relations
News/Press
Company Philosophy
Jobs
Partners
step 1
FIRST NAME
xendit
LAST NAME
tester
EMAIL
xendittester100@hotmail.cc
what email address do you want to encryption enable?
The Art of Encryption
(n-kript)
Providing World Class e-confidentiality solutions
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Internet

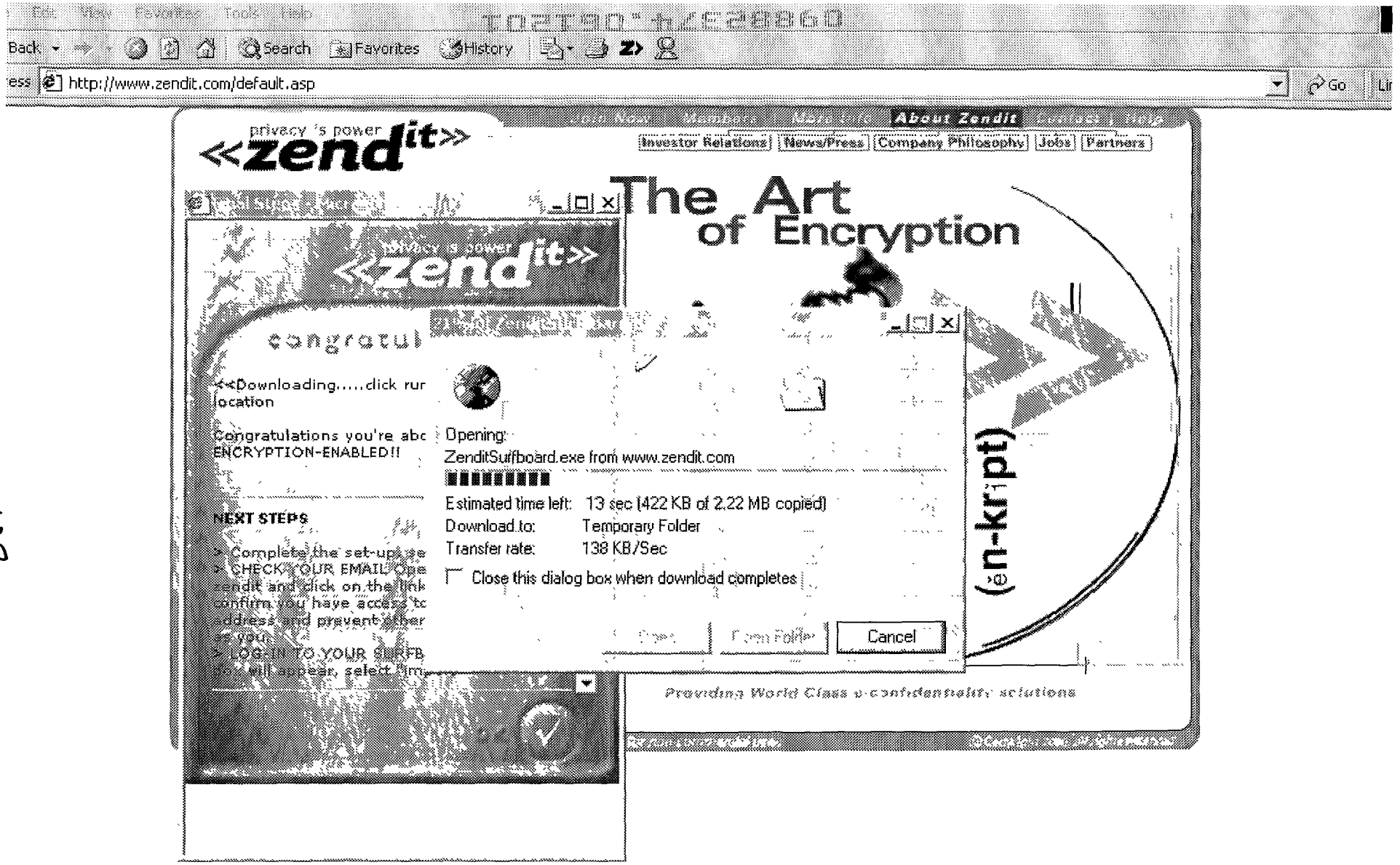
123


Back
Search
Favorites
History
http://www.zendit.com/default.asp
Go
privacy is power
zendit
Investor Relations
News/Press
Company Philosophy
Jobs
Partners
step 2
USER NAME
xendittester100
This can be any name of your choosing and will work with your Zendit Vault and Surfboard.
PASSWORD
This is to ensure privacy and security.
Re-ENTER PASSWORD
The Art of Encryption
(e-n-kript)
Providing World Class e-confidentiality solutions.
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Internet

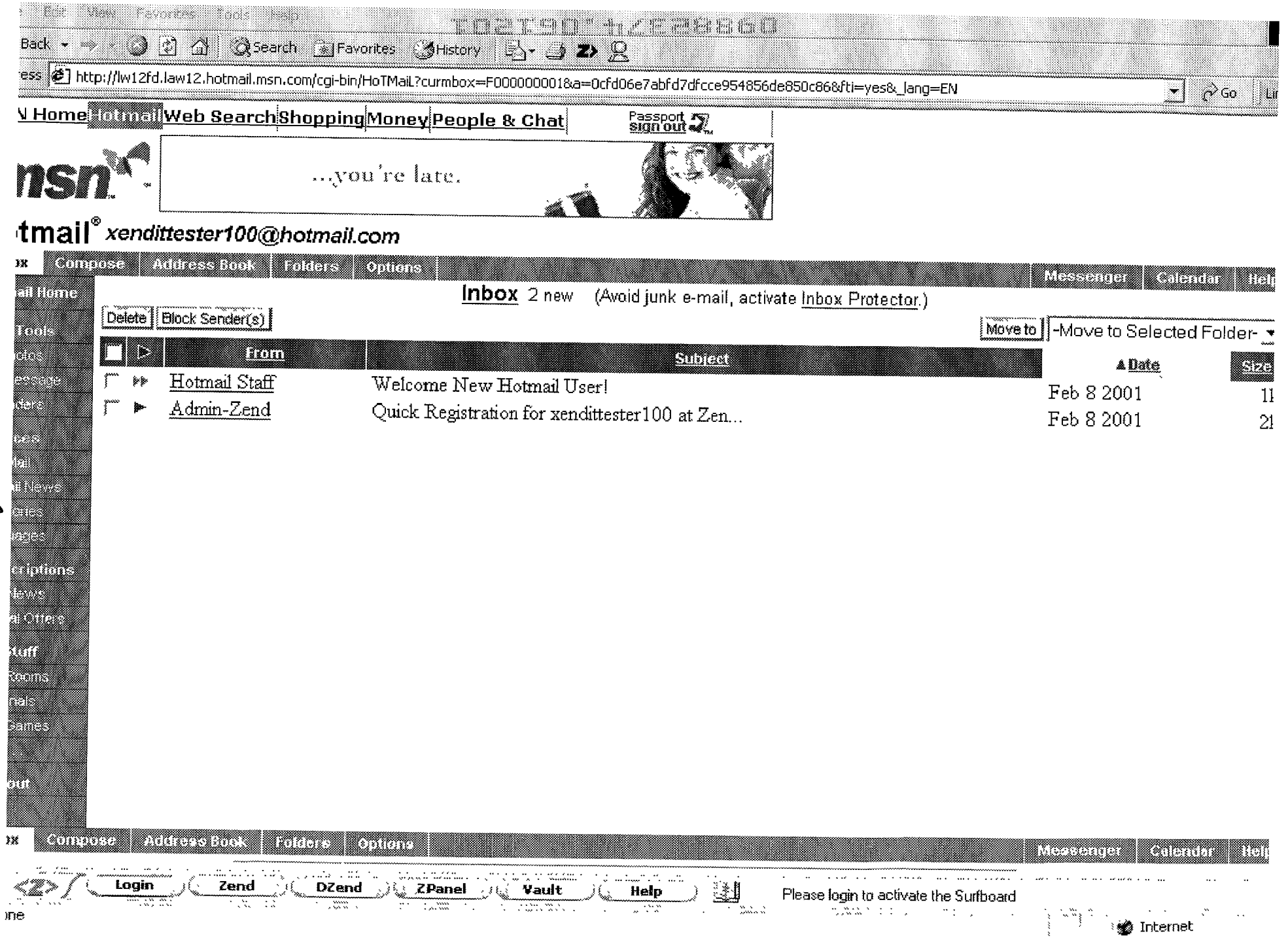

Back
Search
Favorites
History
http://www.zendit.com/default.asp
Go
privacy is power
<<zendit>>
About Zendit
Investor Relations
News/Press
Company Philosophy
Jobs
Partners
The Art of Encryption
(n-kr1pt)
Providing World Class e-confidentiality solutions.
privacy is power
<<zendit>>
Congratulations!
<<Downloading.....click run from current location
Congratulations you're about to become ENCRYPTION-ENABLED!!
NEXT STEPS
> Complete the set-up, select finish
> CHECK YOUR EMAIL Open the email from zendit and click on the link inside. This will confirm you have access to the email address and prevent others masquerading as you.
> LOG-IN TO YOUR SURFBOARD A message will appear, select "import"
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Internet


Back
Search
Favorites
History
http://www.zendit.com/default.asp
Go
privacy is power
zendit
About Zendit
Investor Relations
News/Press
Company Philosophy
Jobs
Partners
The Art of Encryption
File Download
You have chosen to download a file from this location.
ZenditSurfboard.exe from www.zendit.com
What would you like to do with this file?
Run this program from its current location
Save this program to disk
OK
Cancel
More Info
Congra
<<Downloading.....cl
location
Congratulations you'
ENCRYPTION-ENABLE
NEXT STEPS
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Internet

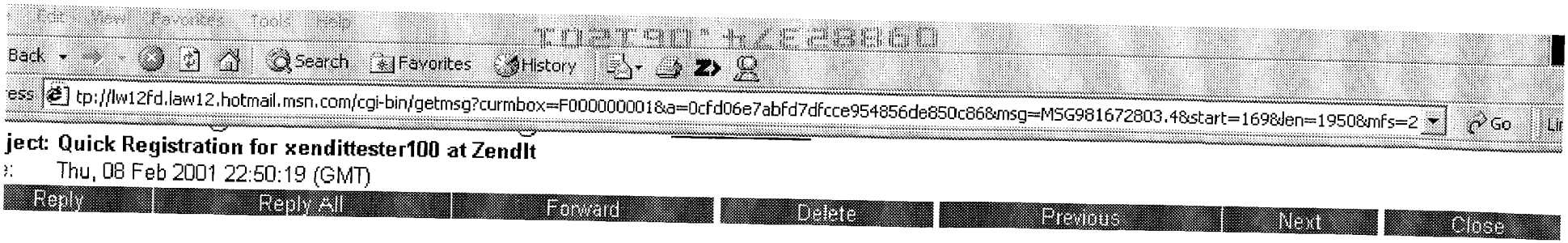
126


Back
Search
Favorites
History
http://www.zendit.com/default.asp
Go
privacy is power
zendit
About Zendit
Investor Relations
News/Press
Company Philosophy
Jobs
Partners
The Art of Encryption
(n-kr¡pt)
congratul
<<Downloading.....click rur location
Congratulations you're abc ENCRYPTION-ENABLED!!
NEXT STEPS
Opening:
ZenditSurfboard.exe from www.zendit.com
Estimated time left: 13 sec (422 KB of 2.22 MB copied)
Download to: Temporary Folder
Transfer rate: 138 KB/Sec
Close this dialog box when download completes
Cancel
Providing World Class e-confidentiality solutions
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Internet

127


Back
Search
Favorites
History
http://lw12fd.law12.hotmail.msn.com/cgi-bin/HoTMaiL?curmbox=F000000001&a=0cfd06e7abfd7dfcce954856de850c86&fti=yes&_lang=EN
Go
Home
Hotmail
Web Search
Shopping
Money
People & Chat
Passport
sign out
msn
...you're late.
tmail® xenditester100@hotmail.com
Compose
Address Book
Folders
Options
Messenger
Calendar
Inbox 2 new (Avoid junk e-mail, activate Inbox Protector.)
Delete
Block Sender(s)
Move to
-Move to Selected Folder-
From
Subject
Date
Size
Hotmail Staff
Welcome New Hotmail User!
Feb 8 2001
Admin-Zend
Quick Registration for xenditester100 at Zen...
Feb 8 2001
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Internet

128



< < < Z E N D I T C O N F I R M A T I O N > > > > >

nk you for registering with Zendit!

C O N F I R M O W N E R S H I P
confirm your ownership of this email address please click on the link below:

p://www.zendit.com/quickconf.asp?LoginID=xendittester100&ConfID=XTLJLHOGAVULUXXKQFPB&Email=xendittester100^hotmail.com

s is part of a confirmation process that protects you in case someone
empts to impersonate you and claims ownership of your account.

you did not register with Zendit, please do not confirm it.

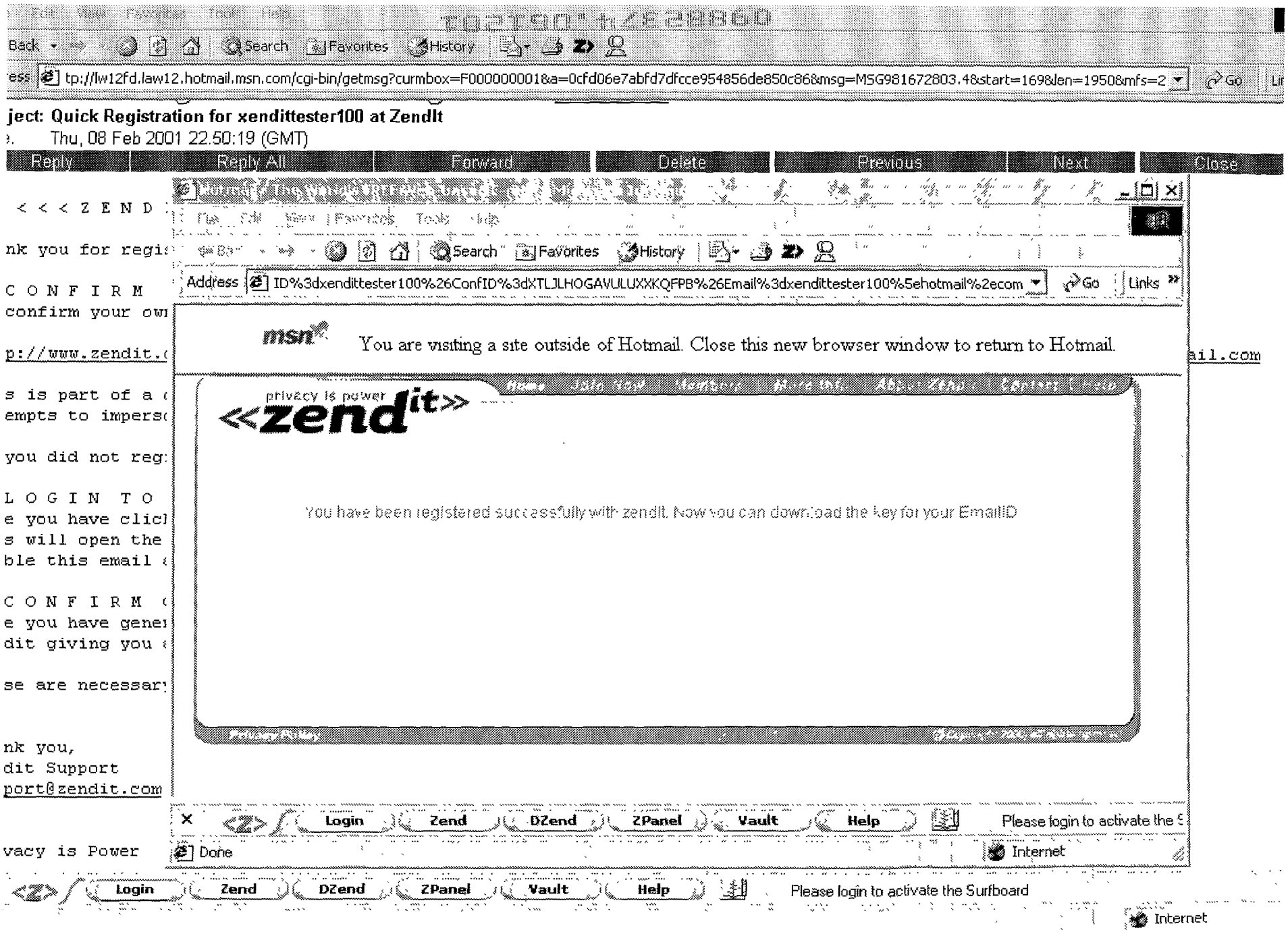
L O G I N T O S U R F B O A R D
e you have clicked on the above link, go ahead and login to the surfboard.
s will open the lock and key/certificate generator that will encryption
ble this email address.

C O N F I R M C E R T I F I C A T E
e you have generated your certificate you will receive one more email from
dit giving you a link to confirm your certificate.

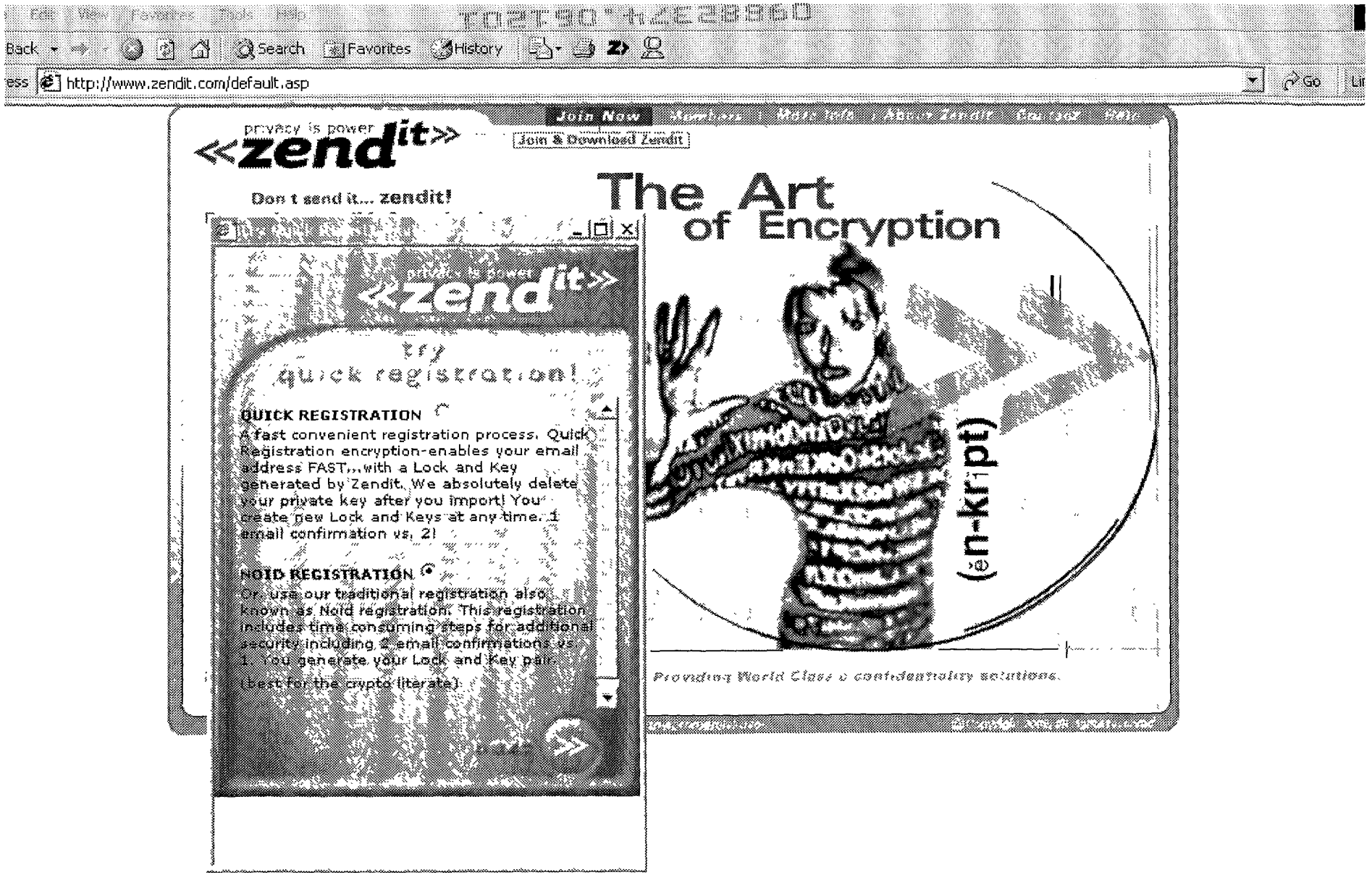
se are necessary steps to ensure the integrity of your certificate.

nk you,
dit Support
port@zendit.com

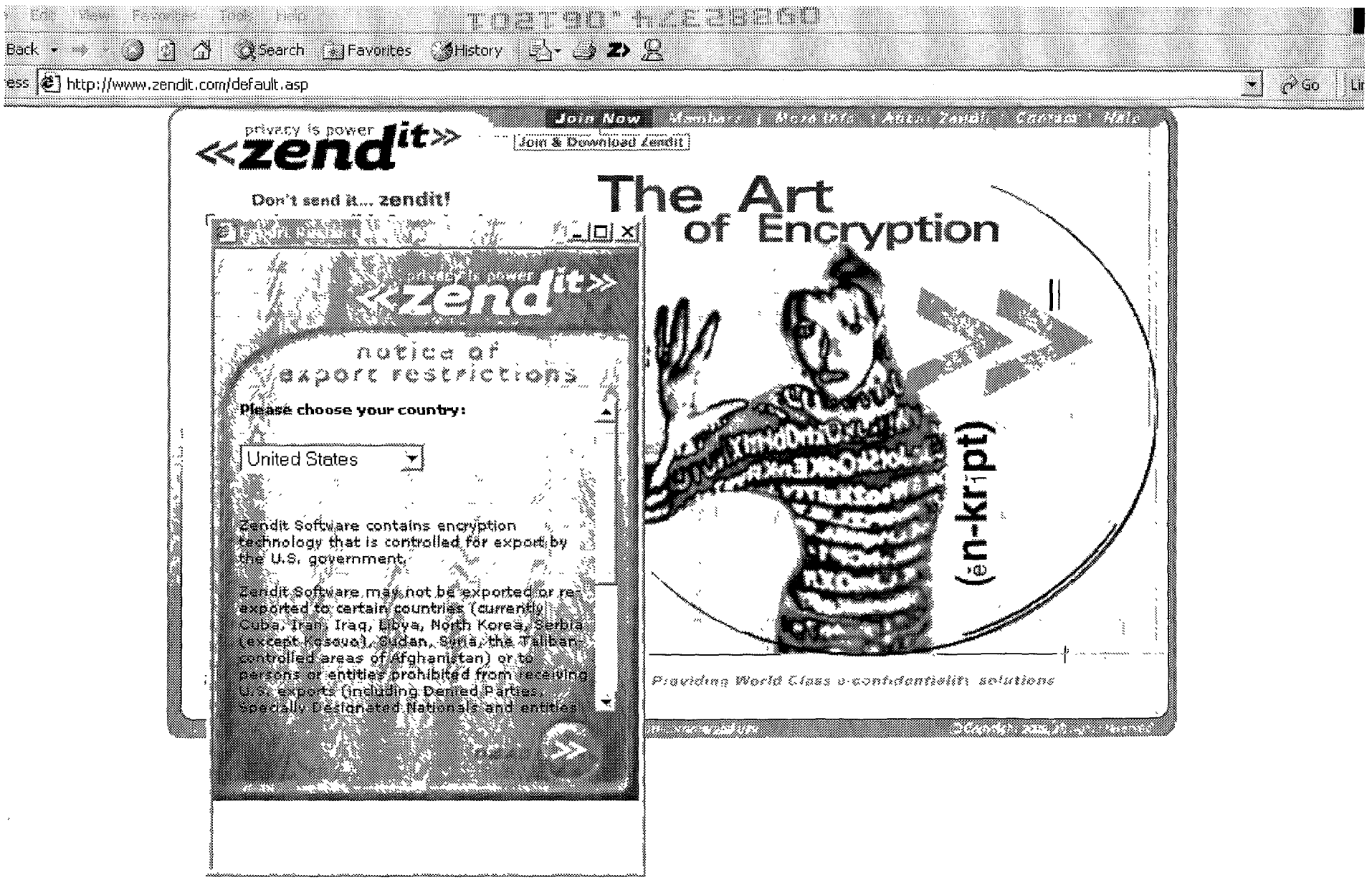
vacy is Power

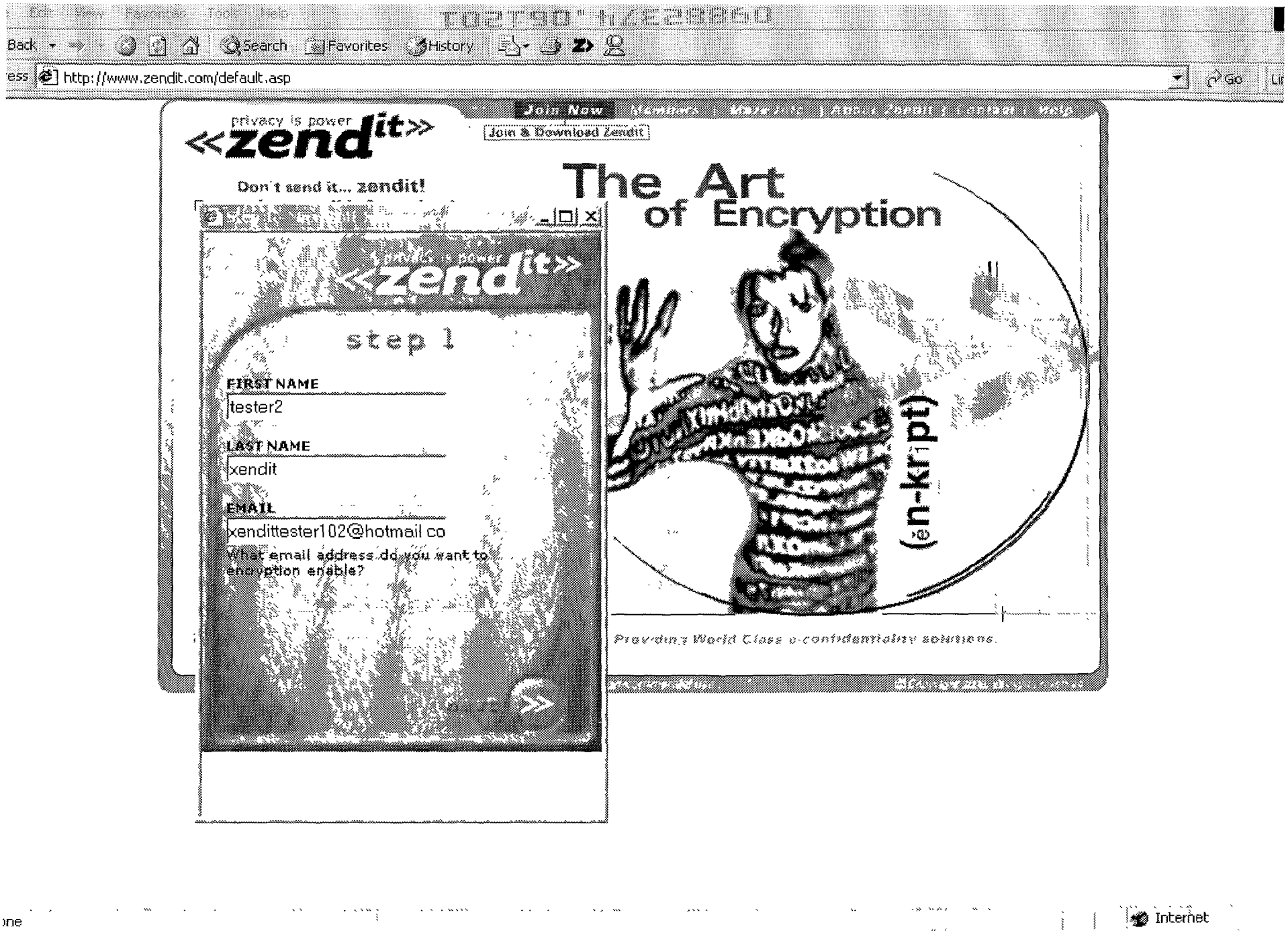

Back
Search
Favorites
History
tp://lw12fd.law12.hotmail.msn.com/cgi-bin/getmsg?curmbox=F000000001&a=0cfd06e7abfd7dfcce954856de850c86&msg=MSG981672803.48&start=169&len=1950&mfs=2
Go
ject: Quick Registration for xendittester100 at Zendlt
Thu, 08 Feb 2001 22.50:19 (GMT)
Reply
Reply All
Forward
Delete
Previous
Next
Close
< < < Z E N D
nk you for regi
C O N F I R M
confirm your ow
p://www.zendit.
s is part of a
empts to impers
you did not reg
L O G I N T O
e you have clic
s will open the
ble this email
C O N F I R M
e you have gene
dit giving you
se are necessar
nk you,
dit Support
port@zendit.com
vacy is Power
Address
ID%3dxendittester100%26ConfID%3dXTLJLHOGAVULUXXKQFPB%26Email%3dxendittester100%5ehotmail%2ecom
Links
msn
You are visiting a site outside of Hotmail. Close this new browser window to return to Hotmail.
privacy is power
zendit
You have been registered successfully with zendit. Now you can download the key for your EmailID
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Done
Internet

Noid Registration

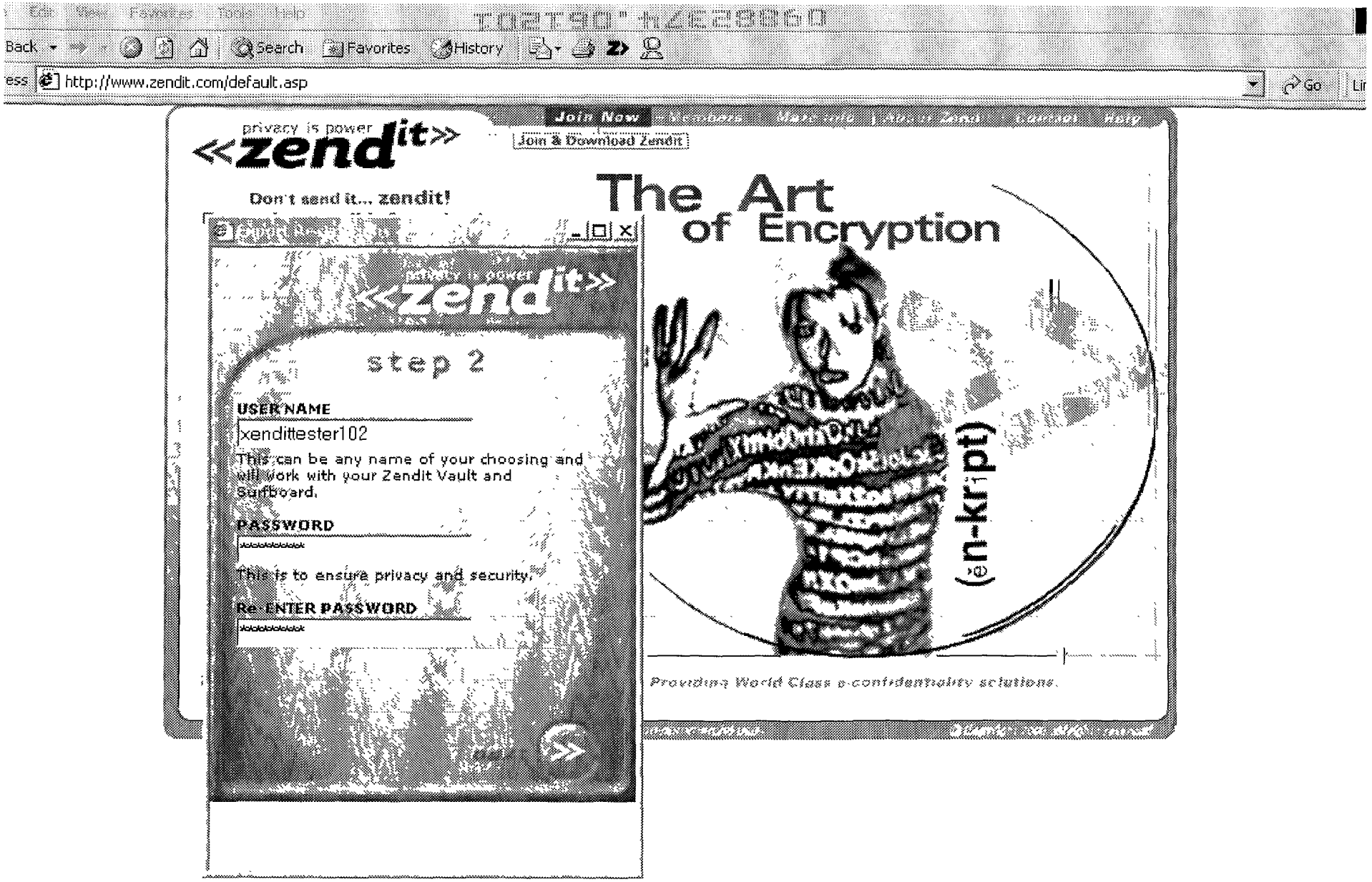

Back
Search
Favorites
History
http://www.zendit.com/default.asp
Go
privacy is power
zendit
Join Now
Join & Download Zendit
Don t send it... zendit!
The Art
of Encryption
(n-kript)
try
quick registration!
QUICK REGISTRATION
A fast convenient registration process. Quick Registration encryption-enables your email address FAST...with a Lock and Key generated by Zendit. We absolutely delete your private key after you import! You create new Lock and Keys at any time. 1 email confirmation vs. 2!
NOID REGISTRATION
Or, use our traditional registration also known as Noid registration. This registration includes time consuming steps for additional security including 2 email confirmations vs. 1. You generate your Lock and Key pair.
(best for the crypto literate)
Providing World Class e confidentiality solutions.
Internet

(32

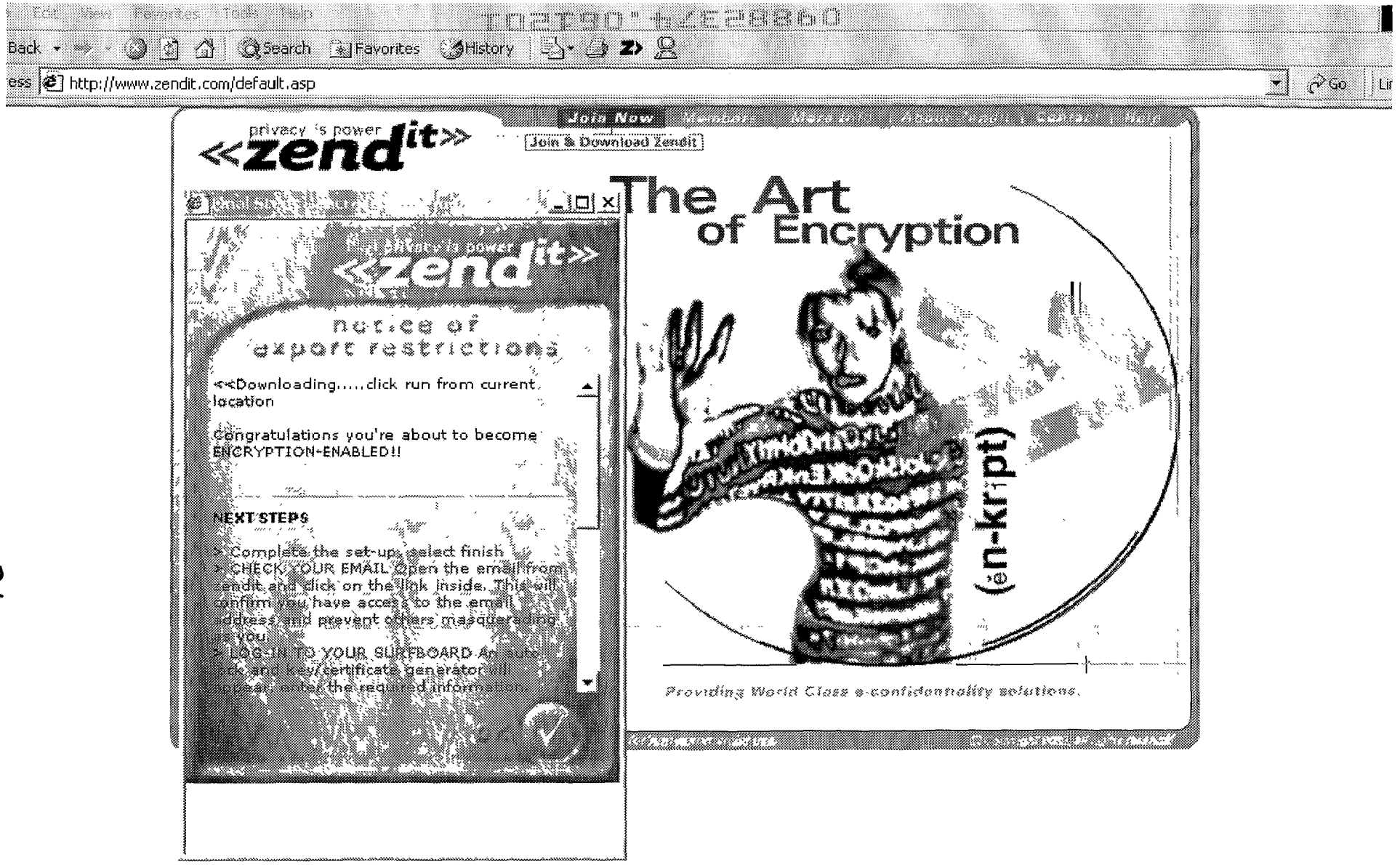

Back
Search
Favorites
History
http://www.zendit.com/default.asp
Go
privacy is power
zendit
Join Now
Join & Download Zendit
Don't send it... zendit!
The Art of Encryption
notice of export restrictions
Please choose your country:
United States
Zendit Software contains encryption technology that is controlled for export by the U.S. government.
Zendit Software may not be exported or re-exported to certain countries (currently Cuba, Iran, Iraq, Libya, North Korea, Serbia (except Kosovo), Sudan, Syria, the Taliban-controlled areas of Afghanistan) or to persons or entities prohibited from receiving U.S. exports (including Denied Parties, Specially Designated Nationals and entities
(e-kript)
Providing World Class e-confidentiality solutions
Internet

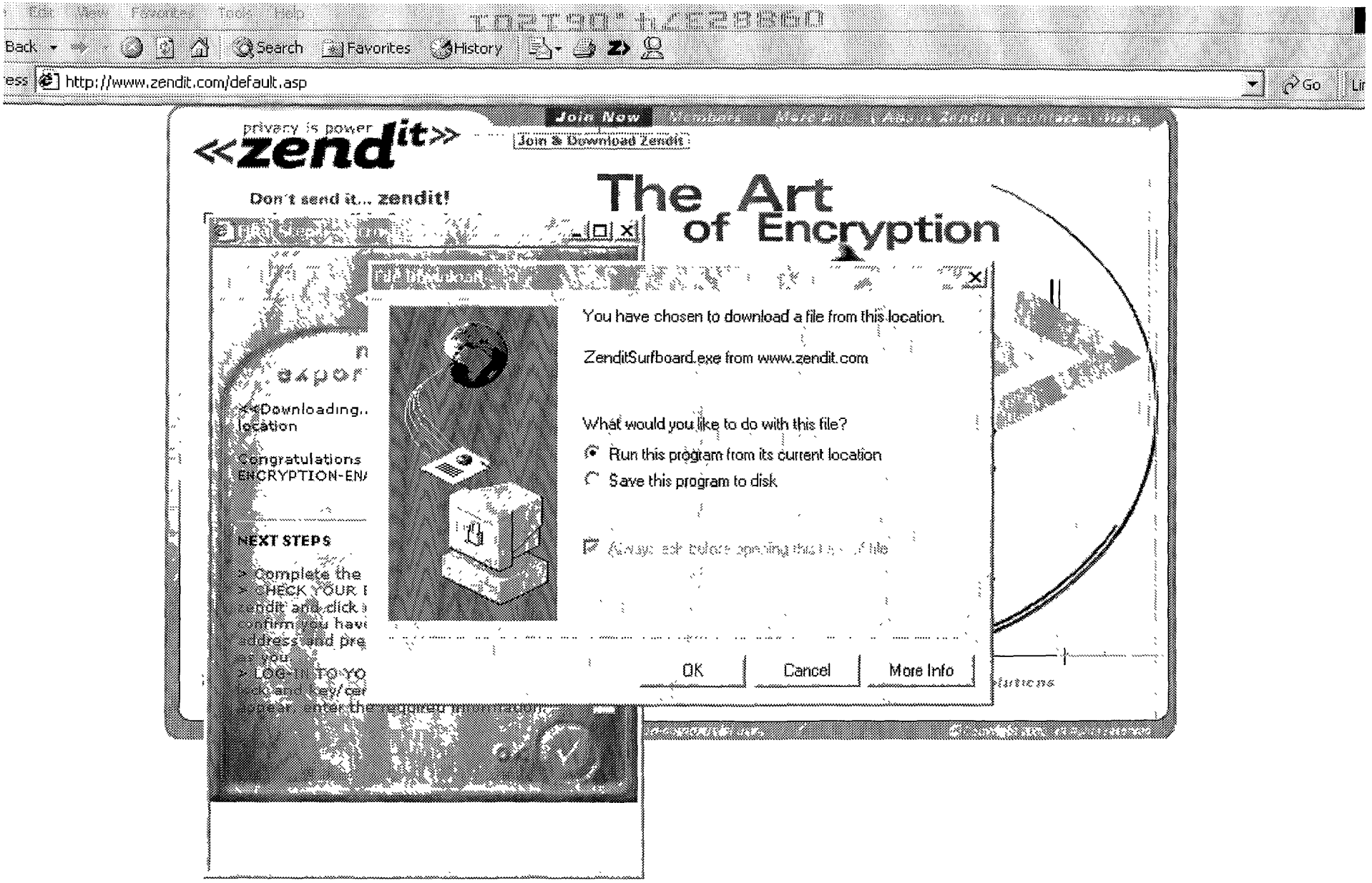

Back
Search
Favorites
History
http://www.zendit.com/default.asp
Go
privacy is power
zendit
Join Now
Join & Download Zendit
Don't send it... zendit!
The Art
of Encryption
(en-krip-pt)
step 1
FIRST NAME
tester2
LAST NAME
xendit
EMAIL
xenditester102@hotmail.co
What email address do you want to encryption enable?
Internet

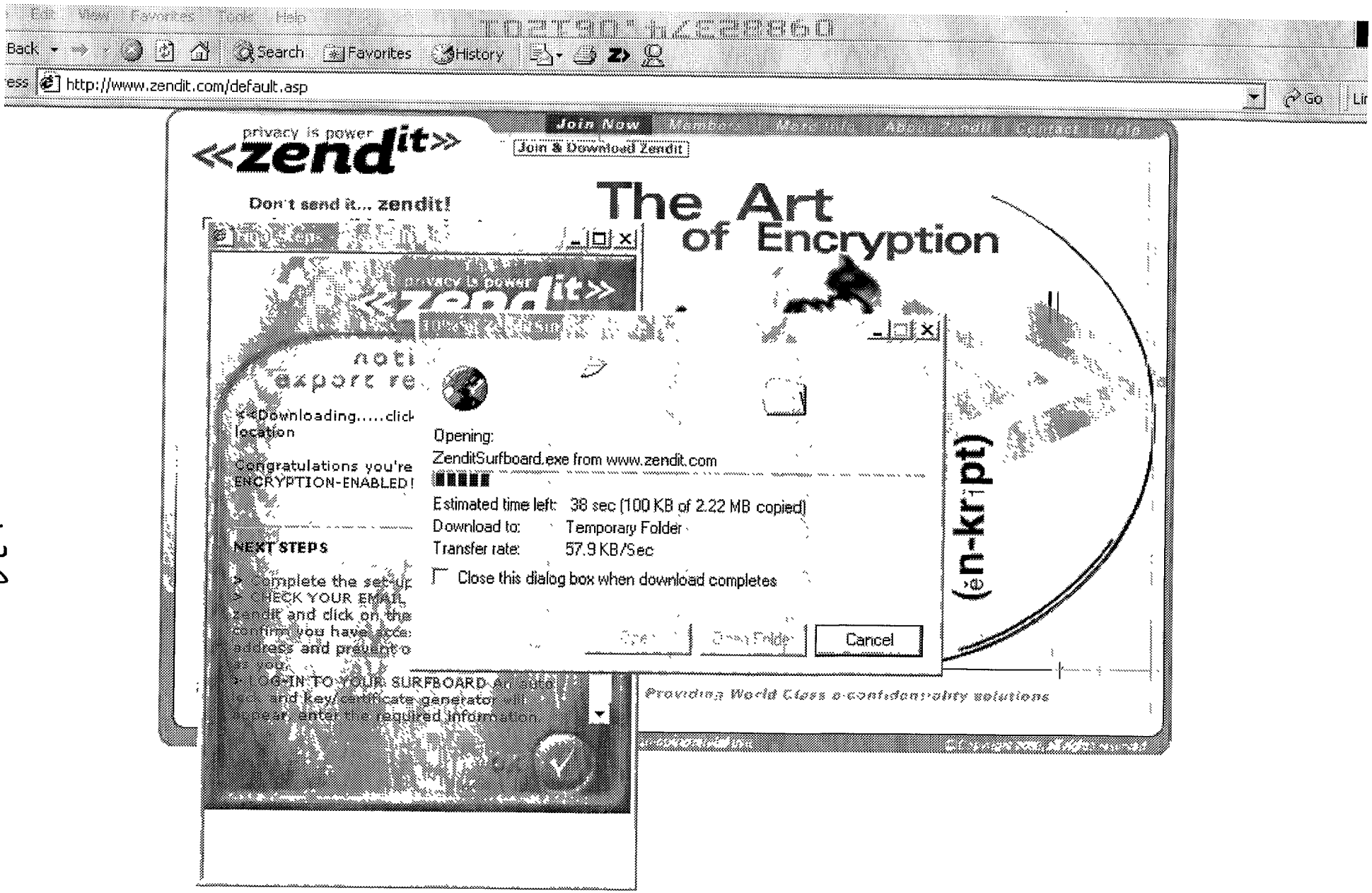
134


Back
Search
Favorites
History
http://www.zendit.com/default.asp
Go
privacy is power
zendit
Join Now
Join & Download Zendit
Don't send it... zendit!
The Art of Encryption
(en-kr1pt)
Providing World Class e-confidentiality solutions.
step 2
USER NAME
xenditester102
This can be any name of your choosing and will work with your Zendit Vault and Surfboard.
PASSWORD
This is to ensure privacy and security.
Re-ENTER PASSWORD
Internet

135

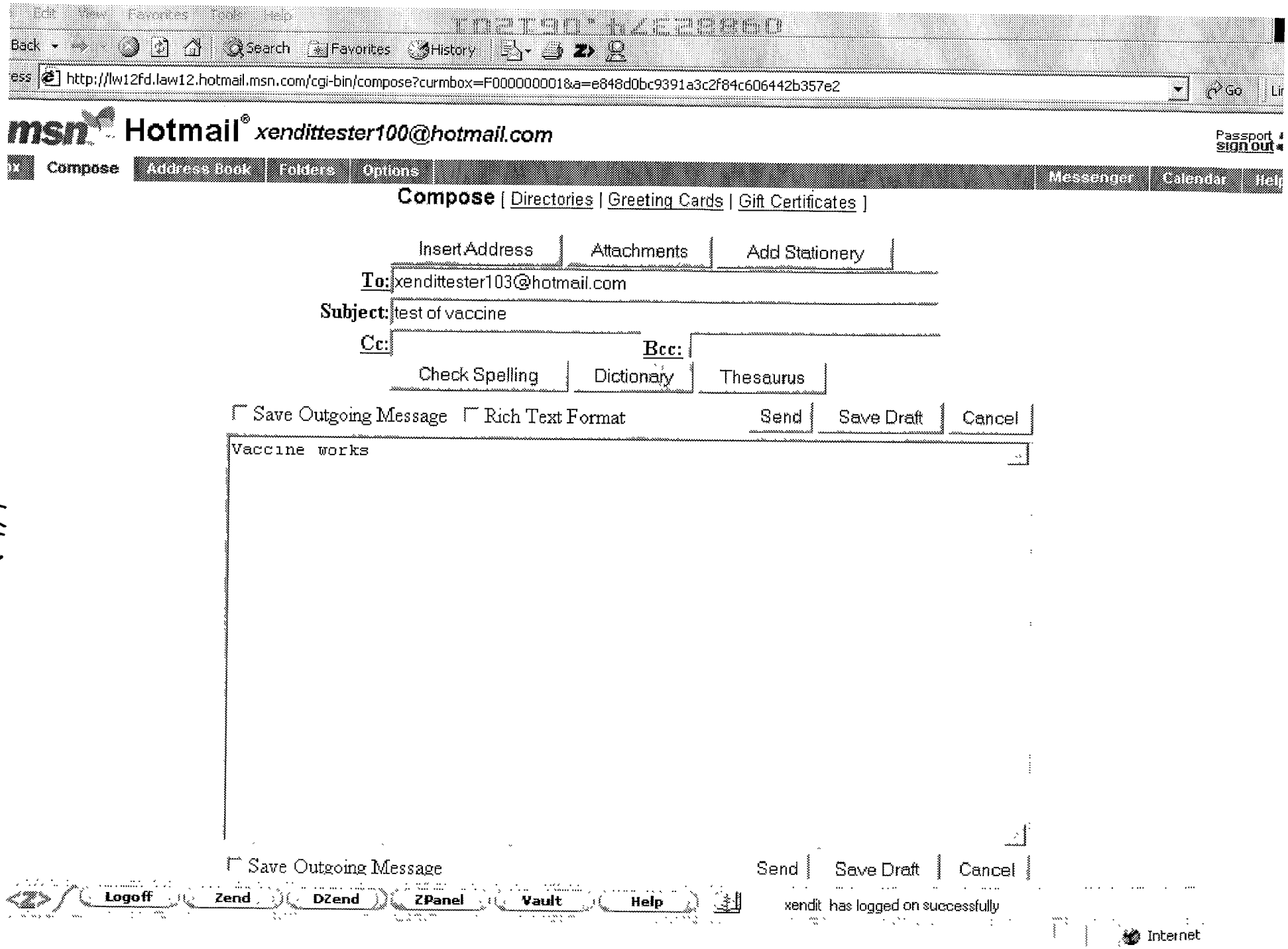

Back
Search
Favorites
History
http://www.zendit.com/default.asp
Go
privacy is power
<<zendit>>
Join Now
Join & Download Zendit
notice of
export restrictions
<<Downloading.....click run from current location
Congratulations you're about to become ENCRYPTION-ENABLED!!
NEXT STEPS
> Complete the set-up, select finish
> CHECK YOUR EMAIL Open the email from zendit and click on the link inside. This will confirm you have access to the email address and prevent others masquerading as you
> LOG-IN TO YOUR SURFBOARD An auto-key and key certificate generator will appear, enter the required information
The Art of Encryption
(n-kr1pt)
Providing World Class e-confidentiality solutions.
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Internet

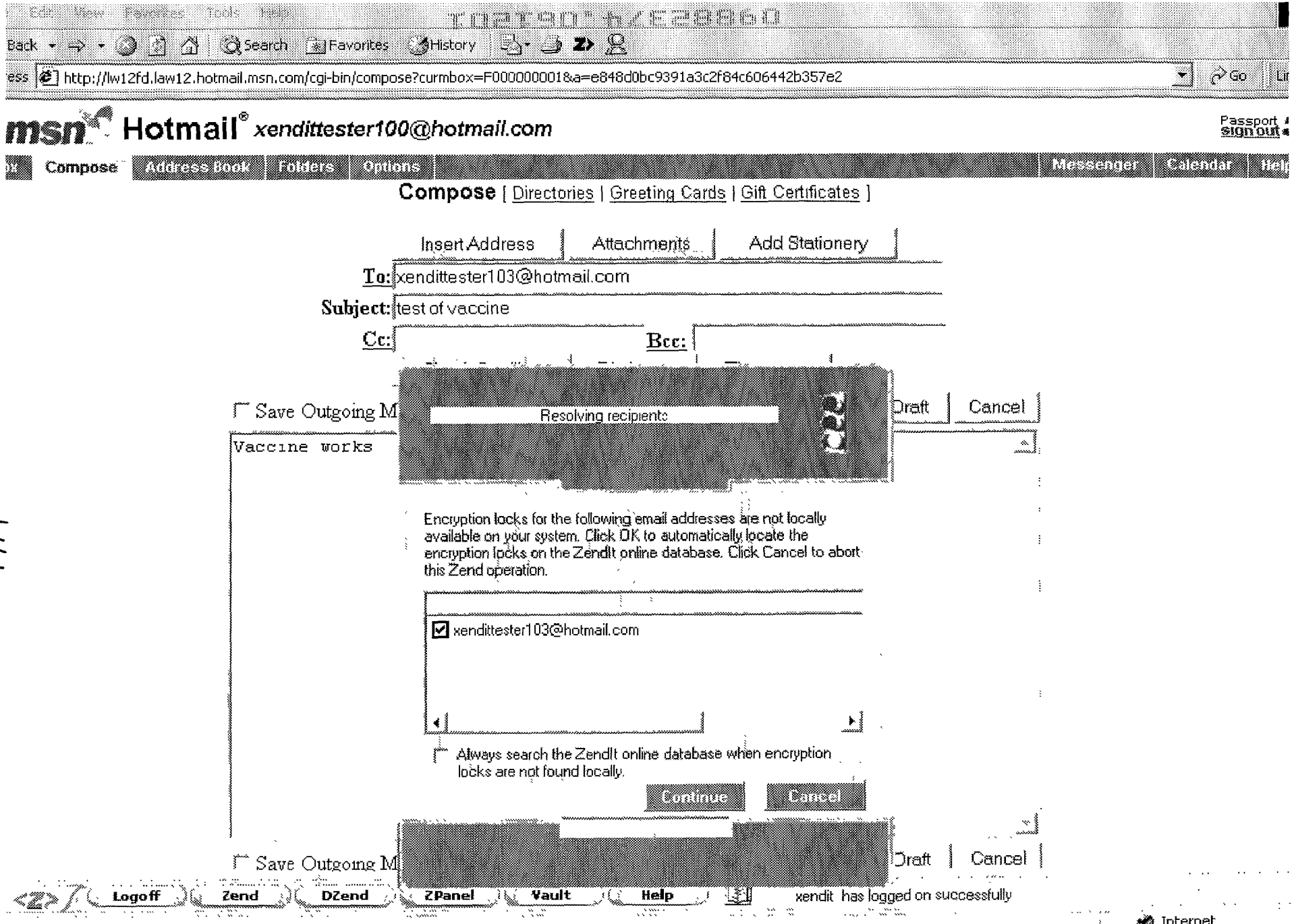

Back
Search
Favorites
History
http://www.zendit.com/default.asp
Go
privacy is power
<<zendit>>
Join Now
Join & Download Zendit
Don't send it... zendit!
The Art of Encryption
You have chosen to download a file from this location.
ZenditSurfboard.exe from www.zendit.com
What would you like to do with this file?
Run this program from its current location
Save this program to disk
OK
Cancel
More Info
NEXT STEPS
Internet


Back
Search
Favorites
History
http://www.zendit.com/default.asp
Go
privacy is power
zendit
Join Now
Join & Download Zendit
Don't send it... zendit!
The Art of Encryption
(n-kr1pt)
Opening:
ZenditSurfboard.exe from www.zendit.com
Estimated time left: 38 sec (100 KB of 2.22 MB copied)
Download to: Temporary Folder
Transfer rate: 57.9 KB/Sec
Close this dialog box when download completes
Cancel
Congratulations you're
ENCRYPTION-ENABLED!
NEXT STEPS
Providing World Class e-confidentiality solutions
Internet

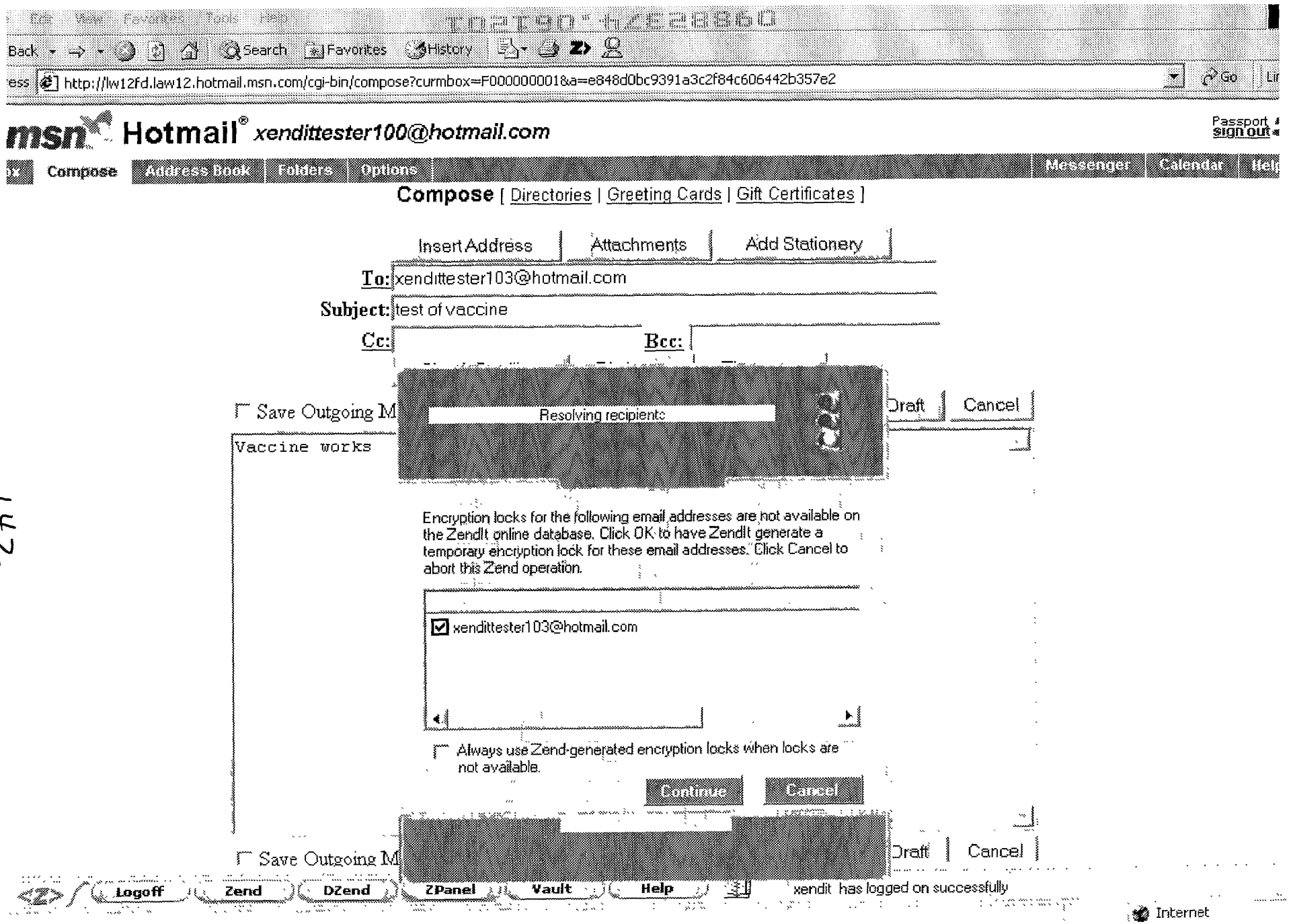
138

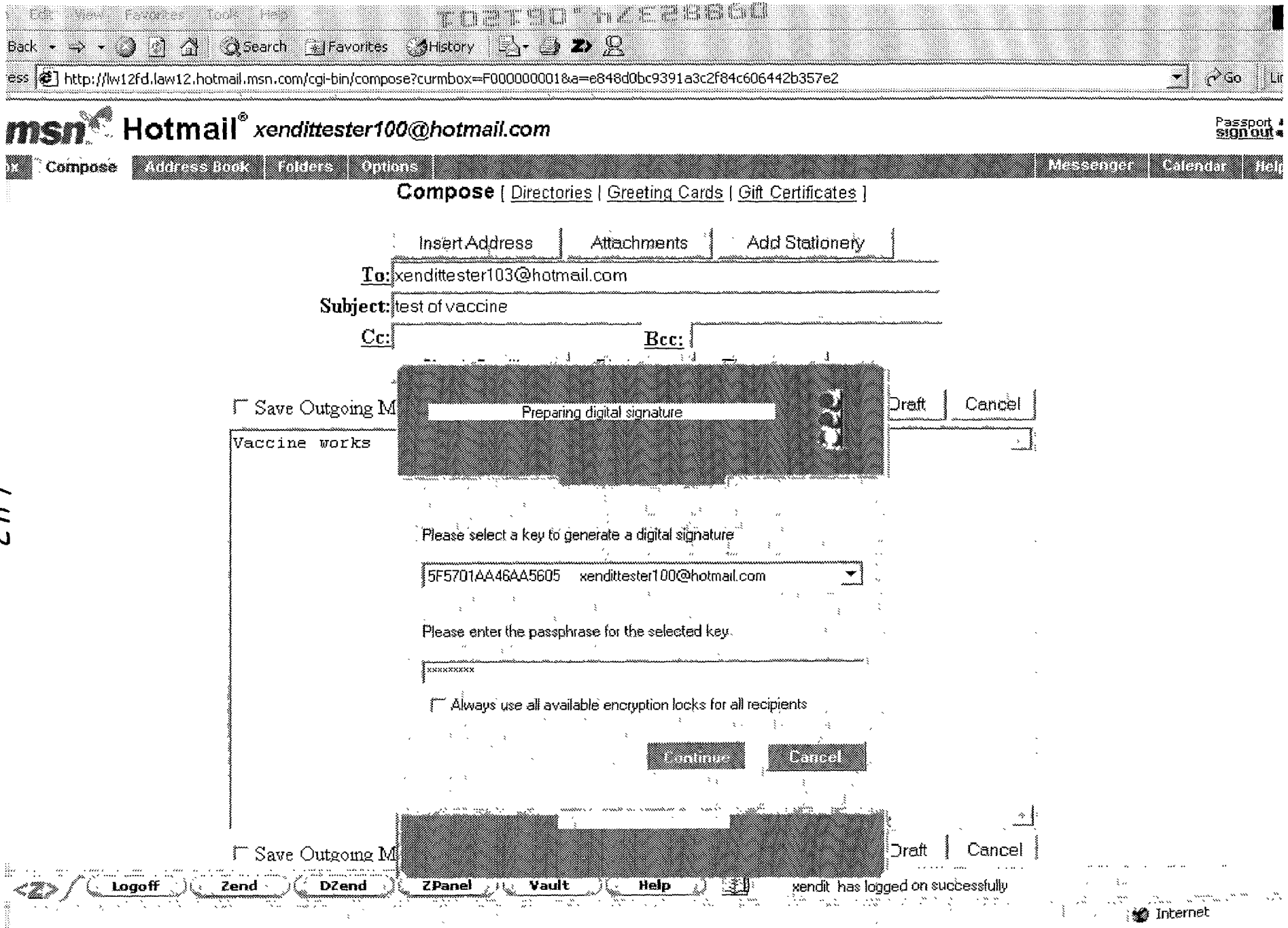
Vaccine Mail

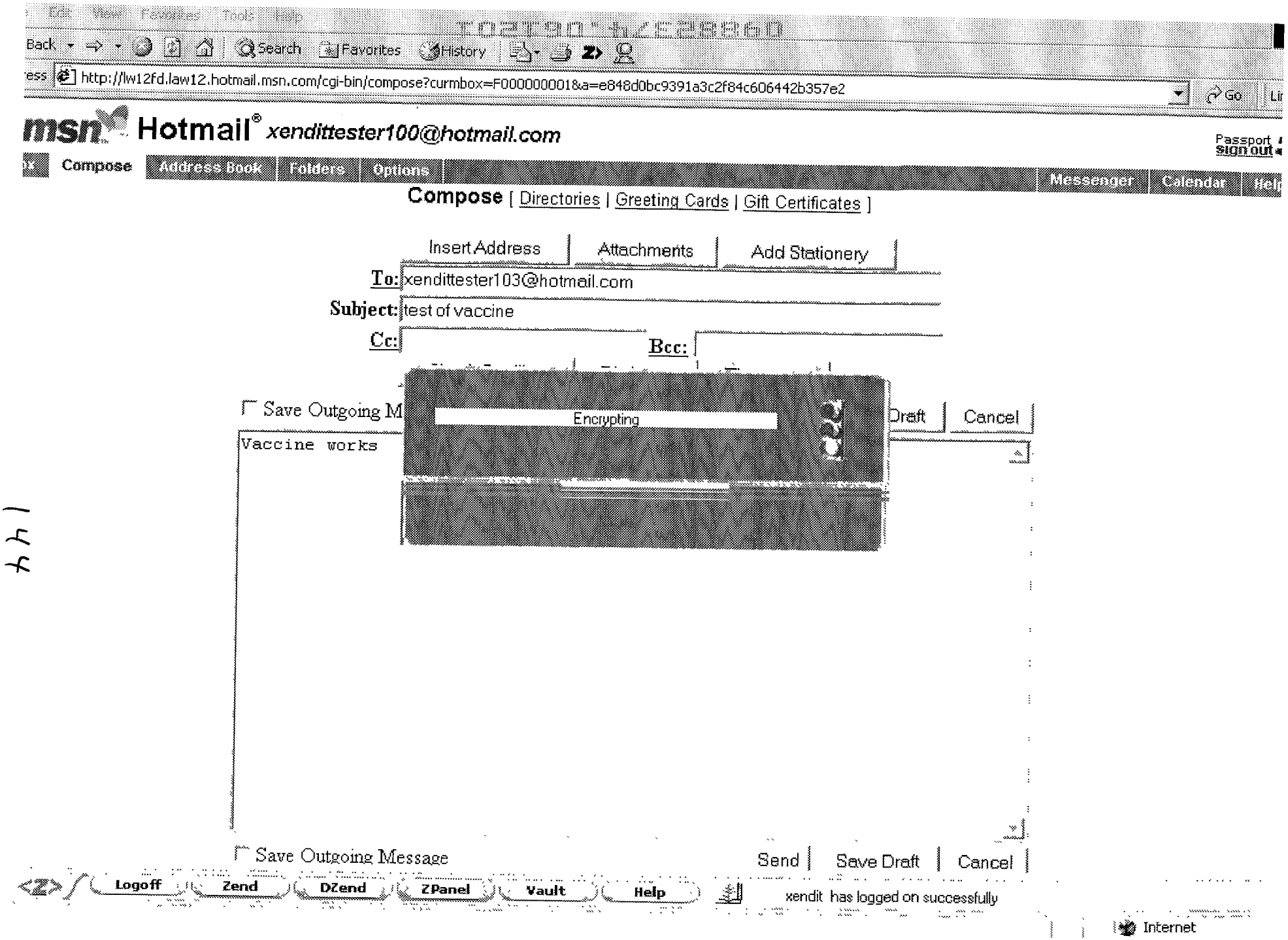

Back
Search
Favorites
History
http://lw12fd.law12.hotmail.msn.com/cgi-bin/compose?curmbox=F000000001&a=e848d0bc9391a3c2f84c606442b357e2
Go
msn Hotmail® xendittester100@hotmail.com
Passport
sign out
Compose
Address Book
Folders
Options
Messenger
Calendar
Compose [Directories | Greeting Cards | Gift Certificates]
Insert Address
Attachments
Add Stationery
To: xendittester103@hotmail.com
Subject: test of vaccine
Cc:
Bcc:
Check Spelling
Dictionary
Thesaurus
Save Outgoing Message
Rich Text Format
Send
Save Draft
Cancel
Vaccine works
Save Outgoing Message
Send
Save Draft
Cancel
Logoff
Zend
DZend
ZPanel
Vault
Help
xendit has logged on successfully
Internet

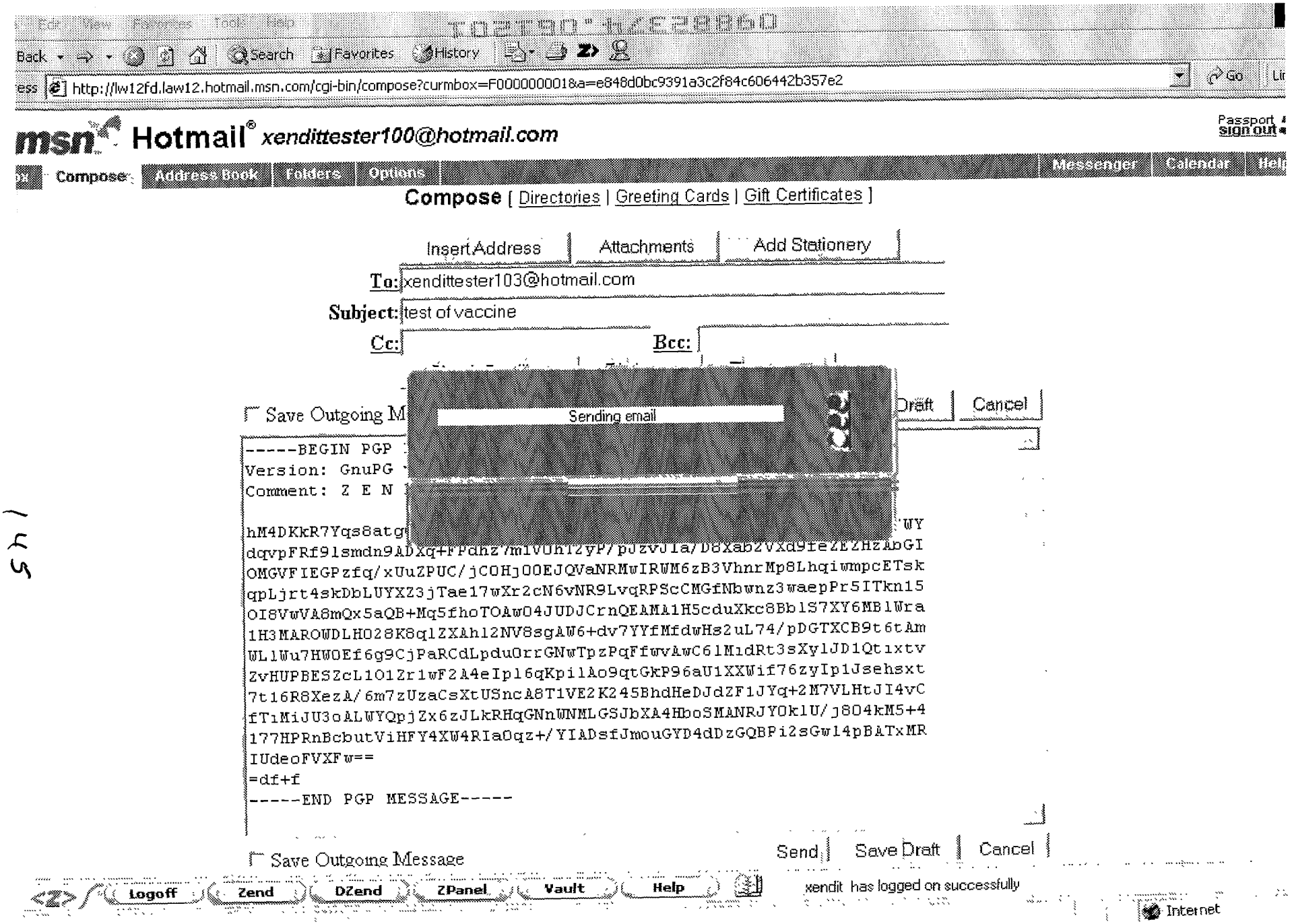

Back
Search
Favorites
History
http://lw12fd.law12.hotmail.msn.com/cgi-bin/compose?curmbox=F000000001&a=e848d0bc9391a3c2f84c606442b357e2
Go
msn Hotmail® xendittester100@hotmail.com
Passport
sign out
Compose
Address Book
Folders
Options
Messenger
Calendar
Compose [Directories | Greeting Cards | Gift Certificates]
Insert Address
Attachments
Add Stationery
To: xendittester103@hotmail.com
Subject: test of vaccine
Cc:
Bcc:
Save Outgoing M
Resolving recipients
Draft
Cancel
Vaccine works
Encryption locks for the following email addresses are not locally available on your system. Click OK to automatically locate the encryption locks on the ZendIt online database. Click Cancel to abort this Zend operation.
xendittester103@hotmail.com
Always search the ZendIt online database when encryption locks are not found locally.
Continue
Cancel
Save Outgoing M
Draft
Cancel
Logoff
Zend
DZend
ZPanel
Vault
Help
xendit has logged on successfully
Internet

(14)


Back
Search
Favorites
History
http://lw12fd.law12.hotmail.msn.com/cgi-bin/compose?curmbox=F000000001&a=e848d0bc9391a3c2f84c606442b357e2
Go
msn Hotmail® xenditester100@hotmail.com
Passport
sign out
Compose
Address Book
Folders
Options
Messenger
Calendar
Compose [Directories | Greeting Cards | Gift Certificates]
Insert Address
Attachments
Add Stationery
To: xenditester103@hotmail.com
Subject: test of vaccine
Cc:
Bcc:
Save Outgoing M
Draft
Cancel
Vaccine works
Resolving recipients
Encryption locks for the following email addresses are not available on the ZendIt online database. Click OK to have ZendIt generate a temporary encryption lock for these email addresses. Click Cancel to abort this Zend operation.
xenditester103@hotmail.com
Always use Zend-generated encryption locks when locks are not available.
Continue
Cancel
Save Outgoing M
Draft
Cancel
Logoff
Zend
DZend
ZPanel
Vault
Help
xendit has logged on successfully
Internet


Back
Search
Favorites
History
http://lw12fd.law12.hotmail.msn.com/cgi-bin/compose?curmbox=F000000001&a=e848d0bc9391a3c2f84c606442b357e2
Go
msn Hotmail® xendittester100@hotmail.com
Passport
sign out
Compose
Address Book
Folders
Options
Messenger
Calendar
Compose [Directories | Greeting Cards | Gift Certificates]
Insert Address
Attachments
Add Stationery
To: xendittester103@hotmail.com
Subject: test of vaccine
Cc:
Bcc:
Save Outgoing M
Draft
Cancel
Vaccine works
Preparing digital signature
Please select a key to generate a digital signature
5F5701AA46AA5605 xendittester100@hotmail.com
Please enter the passphrase for the selected key.
Always use all available encryption locks for all recipients
Continue
Cancel
Save Outgoing M
Draft
Cancel
Logoff
Zend
DZend
ZPanel
Vault
Help
xendit has logged on successfully
Internet


Back
Search
Favorites
History
http://lw12fd.law12.hotmail.msn.com/cgi-bin/compose?curmbox=F000000001&a=e848d0bc9391a3c2f84c606442b357e2
Go
msn Hotmail® xenditester100@hotmail.com
Passport
sign out
Compose
Address Book
Folders
Options
Messenger
Calendar
Compose [Directories | Greeting Cards | Gift Certificates]
Insert Address
Attachments
Add Stationery
To: xenditester103@hotmail.com
Subject: test of vaccine
Cc:
Bcc:
Save Outgoing M
Encrypting
Draft
Cancel
Vaccine works
Save Outgoing Message
Send
Save Draft
Cancel
Logoff
Zend
DZend
ZPanel
Vault
Help
xendit has logged on successfully
Internet


Back
Search
Favorites
History
http://lw12fd.law12.hotmail.msn.com/cgi-bin/compose?curmbox=F000000001&a=e848d0bc9391a3c2f84c606442b357e2
Go
msn Hotmail xendittester100@hotmail.com
Passport
sign out
Compose
Address Book
Folders
Options
Messenger
Calendar
Compose [Directories | Greeting Cards | Gift Certificates]
Insert Address
Attachments
Add Stationery
To: xendittester103@hotmail.com
Subject: test of vaccine
Cc:
Bcc:
Sending email
Save Outgoing M
Draft
Cancel
-----BEGIN PGP
Version: GnuPG
Comment: Z E N
hM4DKkR7Yqs8atg
dqvpFRf9lsmdn9ADXq+fPdhz7m1VUhT2yP/pJzvJ1a/D8Xab2VXd9reZEZHzAbGI
OMGVFIEGPzfq/xUuZPUC/jCOHjOOEJQVaNRMwIRWM6zB3VhnrMp8LhqiwmpcETsk
qpLjrt4skDbLUYXZ3jTae17wXr2cN6vNR9LvqRPScCMGfNbwnz3waepPr5ITkn15
OI8VwVA8mQx5aQB+Mq5fhoTOAwO4JUDJCrnQEAMA1H5cduXkc8Bb1S7XY6MB1Wra
1H3MAROWDLHO28K8q1ZXAh12NV8sgAW6+dv7YYfMfdwHs2uL74/pDGTXCB9t6tAm
WL1Wu7HWOEf6g9CjPaRCdLpduOrrGNwTpzPqFfwvAwC61M1dRt3sXy1JD1Qt1xtv
ZvHUPBESZcL1O1Zr1wF2A4eIp16qKpilAo9qtGkP96aU1XXWif76zyIp1Jsehsxt
7t16R8XezA/6m7zUzaCsXtUSncA8T1VE2K245BhdHeDJdZF1JYq+2M7VLHtJI4vC
fT1MiJU3oALWYQpjZx6zJLkRHqGNnWNMLGSJbXA4HboSMANRJYOk1U/j8O4kM5+4
177HPRnBcbutViHFY4XW4RIaOqz+/YIADsfJmouGYD4dDzGQBPi2sGw14pBATxMR
IUdeoFVXFw==
=df+f
-----END PGP MESSAGE-----
Save Outgoing Message
Send
Save Draft
Cancel
Logoff
Zend
DZend
ZPanel
Vault
Help
xendit has logged on successfully
Internet

Back | Search | Favorites | History

http://lw12fd.law12.hotmail.msn.com/cgi-bin/compose?curmbox=F000000001&a=e848d0bc9391a3c2f84c606442b357e2

msn Hotmail® xendittester100@hotmail.com

Passport sign out

Compose | Address Book | Folders | Options | Messenger | Calendar | Help

Compose [Directories | Greeting Cards | Gift Certificates]

Insert Address | Attachments | Add Stationery

To: xendittester103@hotmail com

Subject: test of vaccine

Cc: **Bcc:**

Check Spelling | Dictionary | Thesaurus

☐ Save Outgoing Message ☐ Rich Text Format Send | Save Draft | Cancel

```
-----BEGIN PGP MESSAGE-----
Version: GnuPG v1.0.1 (W32)
Comment: Z E N D I T  B E T A - <Z.0.1>

hM4DKkR7Yqs8atgQAwCuWXCOfIUBECb1NnuHEWwMJj9ZZDgJP7vOZhYobuxbTFWY
dqvpFRf91smdn9ADXq+FPdhz7m1VOhT2yP/pJzvJIa/D8Xab2VXd9feZEZHzAbGI
OMGVFIEGPzfq/xUuZPUC/jCOHjOOEJQVaNRMwIRWM6zB3VhnrMp8LhqiwmpcETsk
qpLjrt4skDbLUYXZ3jTae17wXr2cN6vNR9LvqRPScCMGfNbwnz3waepPr5ITkn15
OI8VwVA8mQx5aQB+Mq5fhoTOAwO4JUDJCrnQEAMA1H5cduXkc8Bb1S7XY6MB1Wra
1H3MAROWDLHO28K8qlZXAhl2NV8sgAW6+dv7YYfMfdwHs2uL74/pDGTXCB9t6tAm
WL1Wu7HWOEf6g9CjPaRCdLpduOrrGNwTpzPqFfwvAwC61MidRt3sXylJD1Qtixtv
ZvHUPBESZcL1O1Zr1wF2A4eIp16qKpilAo9qtGkP96aU1XXWif76zyIp1Jsehsxt
7t16R8XezA/6m7zUzaCsXtUSncA8T1VE2K245BhdHeDJdZF1JYq+2M7VLHtJI4vC
fTiM1JU3oALWYQpjZx6zJLkRHqGNnWNMLGSJbXA4HboSMANRJYOklU/j804kM5+4
177HPRnBcbutViHFY4XW4RIaOqz+/YIADsfJmouGYD4dDzGQBPi2sGwl4pBATxMR
IUdeoFVXFw==
=df+f
-----END PGP MESSAGE-----
```

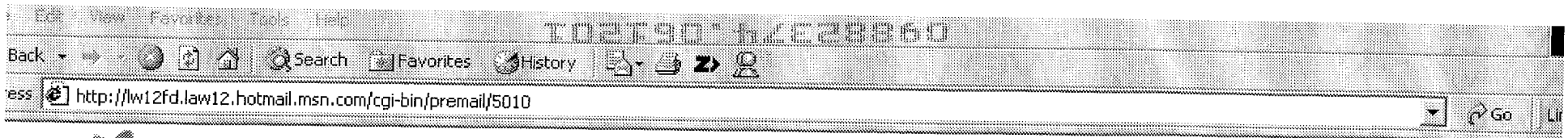
☐ Save Outgoing Message Send | Save Draft | Cancel

Logoff | Zend | DZend | ZPanel | Vault | Help

xendit has logged on successfully

Internet

9 of 11



msn Hotmail® *xendittester100@hotmail.com*

Passport sign out

Compose | Address Book | Folders | Options | Messenger | Calendar | Help

Sent Message Confirmation

Your message has been **instantly delivered** to the following recipients via the **HotmailDirect** messaging system:

xendittester103@hotmail com

OK

My Messenger Buddies
Click here to sign in

Tip: If you do not want to receive this notification, click the **Options** link on the top horizontal menu bar. On the Options page, click the **Preferences** link. Scroll down to the Confirm Sent Messages option and select No.

Compose | Address Book | Folders | Options | Messenger | Calendar | Help

Logoff | Zend | DZend | ZPanel | Vault | Help

xendit has logged on successfully

Internet

Vaccine Registration

Hotmail Inbox - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Search Favorites History

Address http://lw12fd.law12.hotmail.msn.com/cgi-bin/HoTMaiL?curmbox=F000000001&a=a69135b00fb7c1c8382da9f4582dc8d98&fti=yes&_lang=EN Go

MSN Home | Hotmail | Web Search | Shopping | Money | People & Chat | Passport sign out

msn

Do any of these people look familiar?
CLASSMATES.COM

Hotmail® *xenditester103@hotmail.com*

Inbox | Compose | Address Book | Folders | Options | Messenger | Calendar

Hotmail Home
Cool Tools
My Photos
Find Message
Reminders
Services
POP Mail
Hotmail News
Directories
Languages
Subscriptions
Free News
Special Offers
Fun Stuff
Chat Rooms
Personals
Free Games
More
Sign out

Inbox 3 new (Avoid junk e-mail, activate Inbox Protector.)

Delete | Block Sender(s) | Move to | -Move to Selected Folc

	From	Subject	▲Date
▸▸	Hotmail Staff	Welcome New Hotmail User!	Feb 8 2001
▸	xendit test	test of vaccine	Feb 8 2001
▸	Admin-Zend	hotmail has sent an encrypted mail.	Feb 8 2001

Inbox | Compose | Address Book | Folders | Options | Messenger | Calendar

Logoff | Zend | DZend | ZPanel | Vault | Help

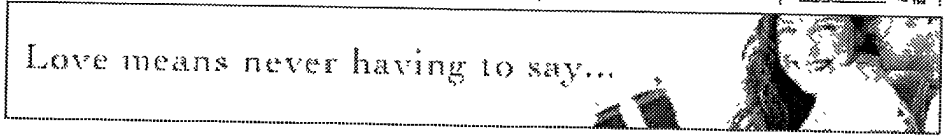
xendit has logged on successfully

6 of 7

Back · Search Favorites History

ess ://lw12fd.law12.hotmail.msn.com/cgi-bin/getmsg?curmbox=F000000001&a=a69135b00fb7c1c8382da9f4582dc8d9&msg=MSG981675172.10&start=1698&len=12338&mfs=2 Go

SN Home | Hotmail | Web Search | Shopping | Money | People & Chat | Passport sign out



otmail® xendittester103@hotmail.com

ox | Compose | Address Book | Folders | Options | Messenger | Calendar | Help

Folder: Inbox

n: **"xendit test" <xendittester100@hotmail.com>** Save Address - Block Sender
xendittester103@hotmail.com Save Address
ject: test of vaccine
: Thu, 08 Feb 2001 23:32:52 -0000

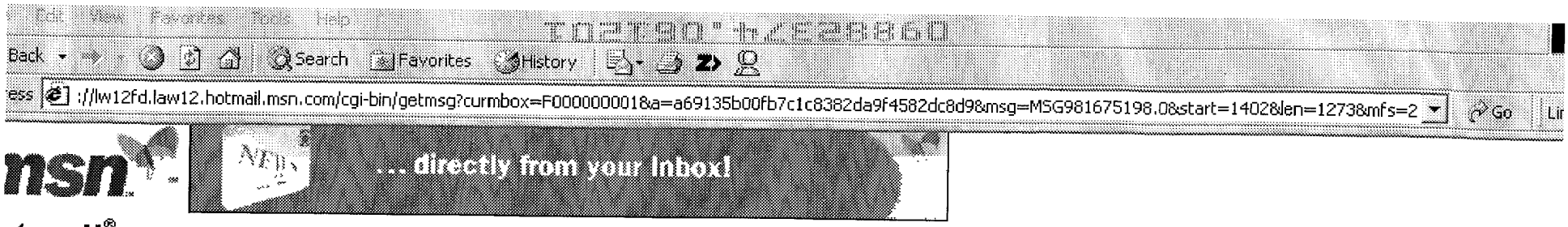
Reply | Reply All | Forward | Delete | Previous | Next | Close

```
--BEGIN PGP MESSAGE-----
sion: GnuPG v1.0.1 (W32)
ment: Z E N D I T  B E T A - <Z.0.1>

DKkR7Yqs8atgQAwCuWXCOfIUBECbiNnuHEWwMJj9ZZDgJP7vOZhYobuxbTFWY
pFRf9lsmdn9ADXq+FPdhz7mlVOhT2yP/pJzvJIa/D8Xab2VXd9feZEZHzAbGI
VFIEGPzfq/xUuZPUC/jCOHjOOEJQVaNRMwIRWM6zB3VhnrMp8LhqiwmpcETsk
jrt4skDbLUYXZ3jTae17wXr2cN6vNR9LvqRPScCMGfNbwnz3waepPr5ITkn15
VwVA8mQx5aQB+Mq5fhoTOAw04JUDJCrnQEAMA1H5cduXkc8Bb1S7XY6MB1Wra
MAROWDLHO28K8q1ZXAh12NV8sgAW6+dv7YYfMfdwHs2uL74/pDGTXCB9t6tAm
Wu7HWOEf6g9CjPaRCdLpduOrrGNwTpzPqFfwvAwC61MidRt3sXylJD1Qtixtv
UPBESZcL1O1Zr1wF2A4eIpl6qKpilAo9qtGkP96aU1XXWif76zyIp1Jsehsxt
6R8XezA/6m7zUzaCsXtUSncA8T1VE2K245BhdHeDJdZF1JYq+2M7VLHtJI4vC
MiJU3oALWYQpjZx6zJLkRHqGNnWNMLGSJbXA4HboSMANRJYOklU/j8O4kM5+4
HPRnBcbutViHFY4XW4RIaOqz+/YIADsfJmouGYD4dDzGQBPi2sGwl4pBATxMR
eoFVXFw==
+f
--END PGP MESSAGE-----
```

Logoff | Zend | DZend | ZPanel | Vault | Help

xendit has logged on successfully

Internet



Folder: Inbox

n: **Admin-Zend <support@zendit.com>** **Save Address** - **Block Sender**
<xenditester103@hotmail.com> **Save Address**
ject: hotmail has sent an encrypted mail.
: Thu, 08 Feb 2001 23:30:00 (GMT)

Reply | Reply All | Forward | Delete | Previous | Next | Close

mail, recently sent you a privileged and confidential email. If you want to
rypt it, please click on the following link to confirm this is YOUR email
ress.

s confirmation protects you in case someone tries to forge ownership of YOUR
il address.

p://www.zendit.com/tempkeywelcome.asp?EmailID=xenditester103^hotmail.com&ConfID=EGGVELDVNUXIRVTCQCKS

nk you,
dit Support
port@zendit.com

vacy is Power

Reply | Reply All | Forward | Delete | Previous | Next | Close

Search Favorites History

http://64.4.18.250/cgi-bin/linkrd?_lang=EN&lah=f86f86b6bcde01e4527bd4d6712d7fbc&lat=9816752608&hm___action=http%3a%2f%2fwww%2ezendit%2ecom%2ftempl Go

msn You are visiting a site outside of Hotmail. Close this new browser window to return to Hotmail.

«zendit»

The Art of Encryption

Welcome to The Zendit Surfboard, a free* encryption software that is fast, simple and easy to use and empowers you to easily encrypt and decrypt e-mail, text and photos.

In order for certain features on the surfboard to work, you need to be a registered user.

If you have not yet signed up online with zendit com please click "next" and we will take you through the process

☐ **I am an existing user**

next->

Zendit is a free service that will help you protect your privacy by making it easy to encrypt the information you send over the Internet. All it takes is a simple tool- the Zendit Surfboard. The Zendit Surfboard lays the foundation for the creation of a new generation of encryption-enabled Internet users who have the means to ensure that the confidentiality of their personal information is protected in cyberspace.

Zendit makes it easy to be "encryption enabled"!

When you download the Zendit Surfboard, it automatically generates what cryptographers refer to as "public key and a "private key " To make the concept a little easier to understand, Zendit prefers to call them . .

Logoff Zend DZend ZPanel Vault Help

xendit has logged on successfully

one

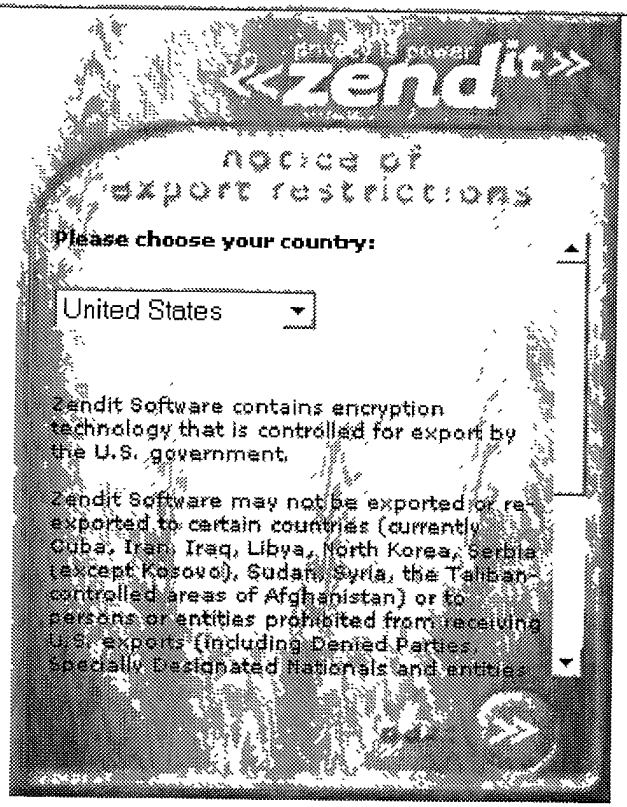
Internet

152

Back Search Favorites History

ess http://64.4.18.250/cgi-bin/linkrd?_lang=EN&lah=f86f86b6bcde01e4527bd4d6712d7fbc&lat=981675260&hm___action=http%3a%2f%2fwww%2ezendit%2ecom%2ftempl Go

msn You are visiting a site outside of Hotmail. Close this new browser window to return to Hotmail



153

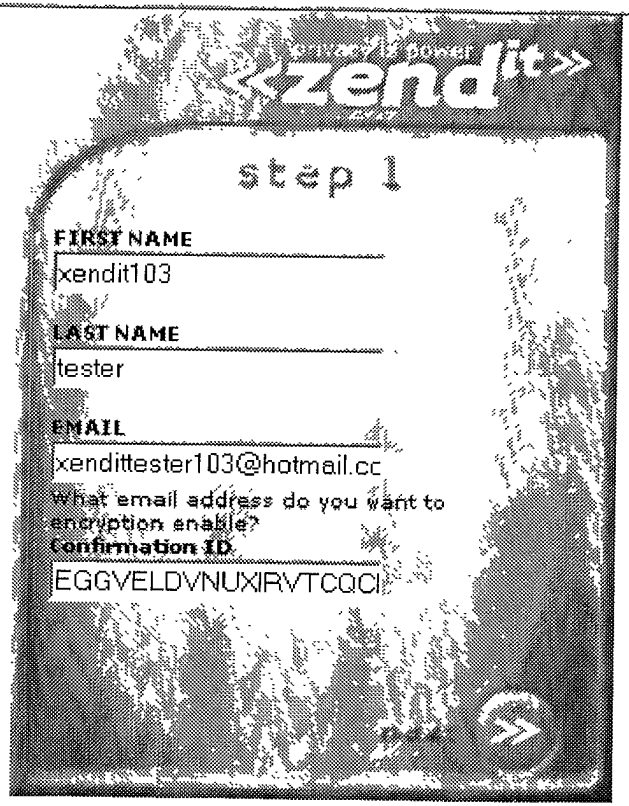
Logoff Zend DZend ZPanel Vault Help xendit has logged on successfully

one Internet

Back Search Favorites History

ess http://64.4.18.250/cgi-bin/linkrd?_lang=EN&lah=f86f86b6bcde01e4527bd4d6712d7fbc&lat=981675260&hm___action=http%3a%2f%2fwww%2ezendit%2ecom%2ftempl Go

msn You are visiting a site outside of Hotmail. Close this new browser window to return to Hotmail.

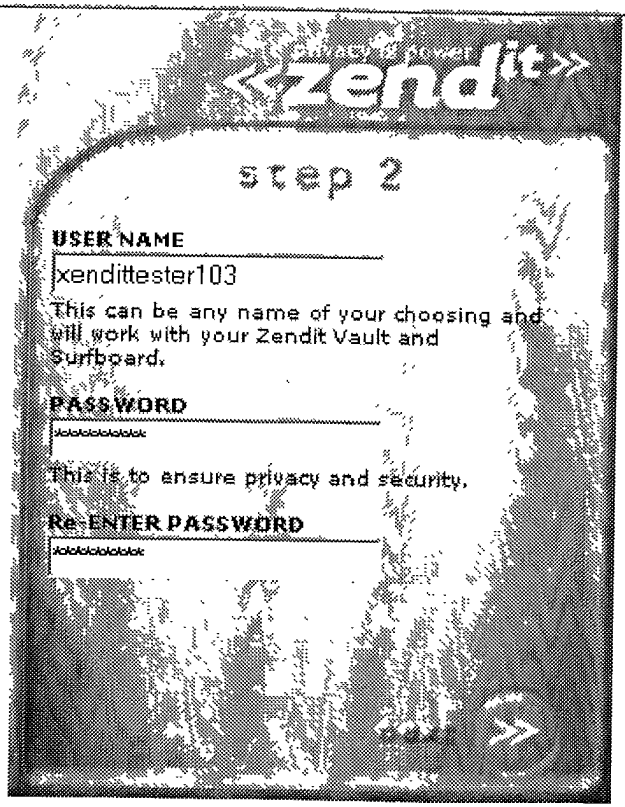


Logoff Zend DZend ZPanel Vault Help

xendit has logged on successfully

ine

Internet

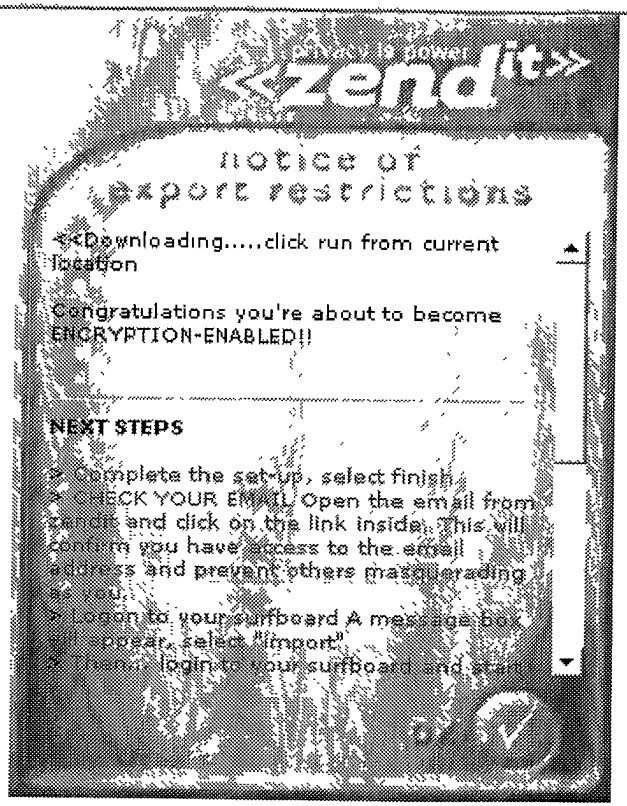


SSL

Back Search Favorites History

http://64.4.18.250/cgi-bin/linkrd?_lang=EN&lah=f86f86b6bcde01e4527bd4d6712d7fbc&lat=981675260&hm___action=http%3a%2f%2fwww%2ezendit%2ecom%2ftempl Go

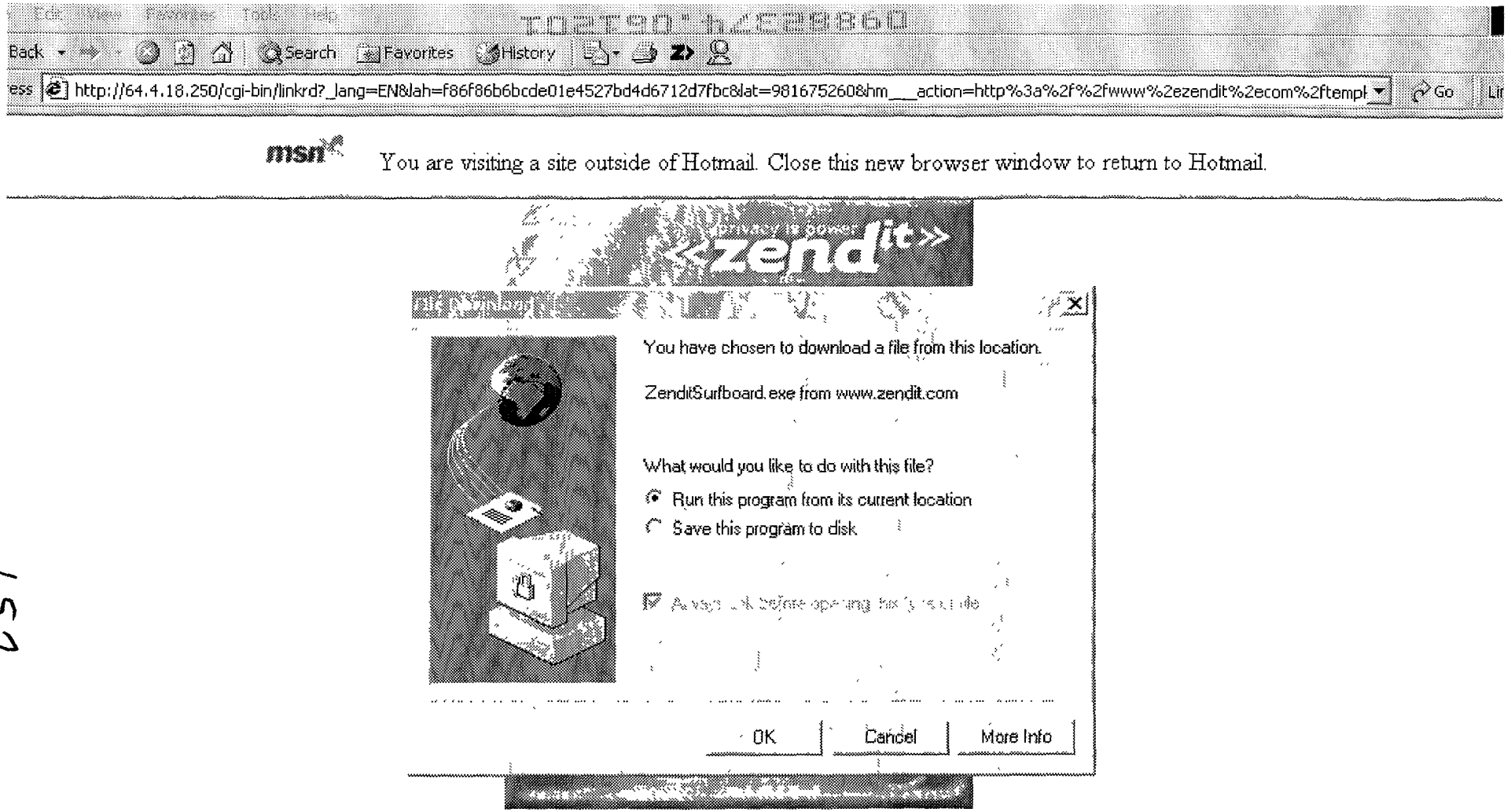
msn You are visiting a site outside of Hotmail. Close this new browser window to return to Hotmail



9 5 1

Logoff Zend DZend ZPanel Vault Help xendit has logged on successfully

Internet

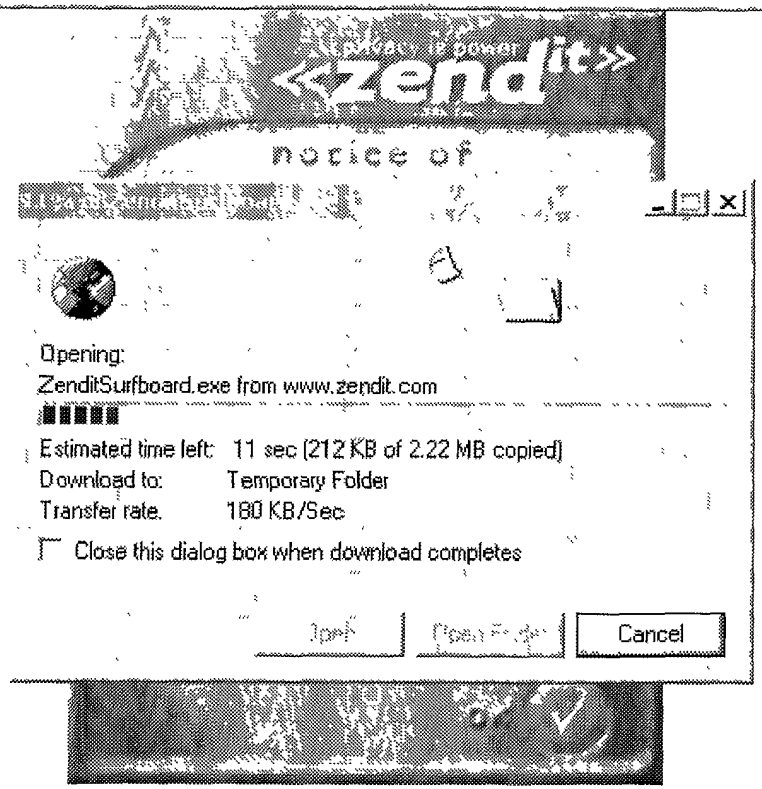

Back
Search
Favorites
History
http://64.4.18.250/cgi-bin/linkrd?_lang=EN&lah=f86f86b6bcde01e4527bd4d6712d7fbc&lat=981675260&hm___action=http%3a%2f%2fwww%2ezendit%2ecom%2ftempl
Go
msn
You are visiting a site outside of Hotmail. Close this new browser window to return to Hotmail.
You have chosen to download a file from this location.
ZenditSurfboard.exe from www.zendit.com
What would you like to do with this file?
Run this program from its current location
Save this program to disk
OK
Cancel
More Info
Logoff
Zend
DZend
ZPanel
Vault
Help
xendit has logged on successfully
Internet

151

Back Search Favorites History

ess http://64.4.18.250/cgi-bin/linkrd?_lang=EN&lah=f86f86b6bcde01e4527bd4d6712d7fbc&lat=9816752608&hm___action=http%3a%2f%2fwww%2ezendit%2ecom%2ftempl Go

msn You are visiting a site outside of Hotmail. Close this new browser window to return to Hotmail.

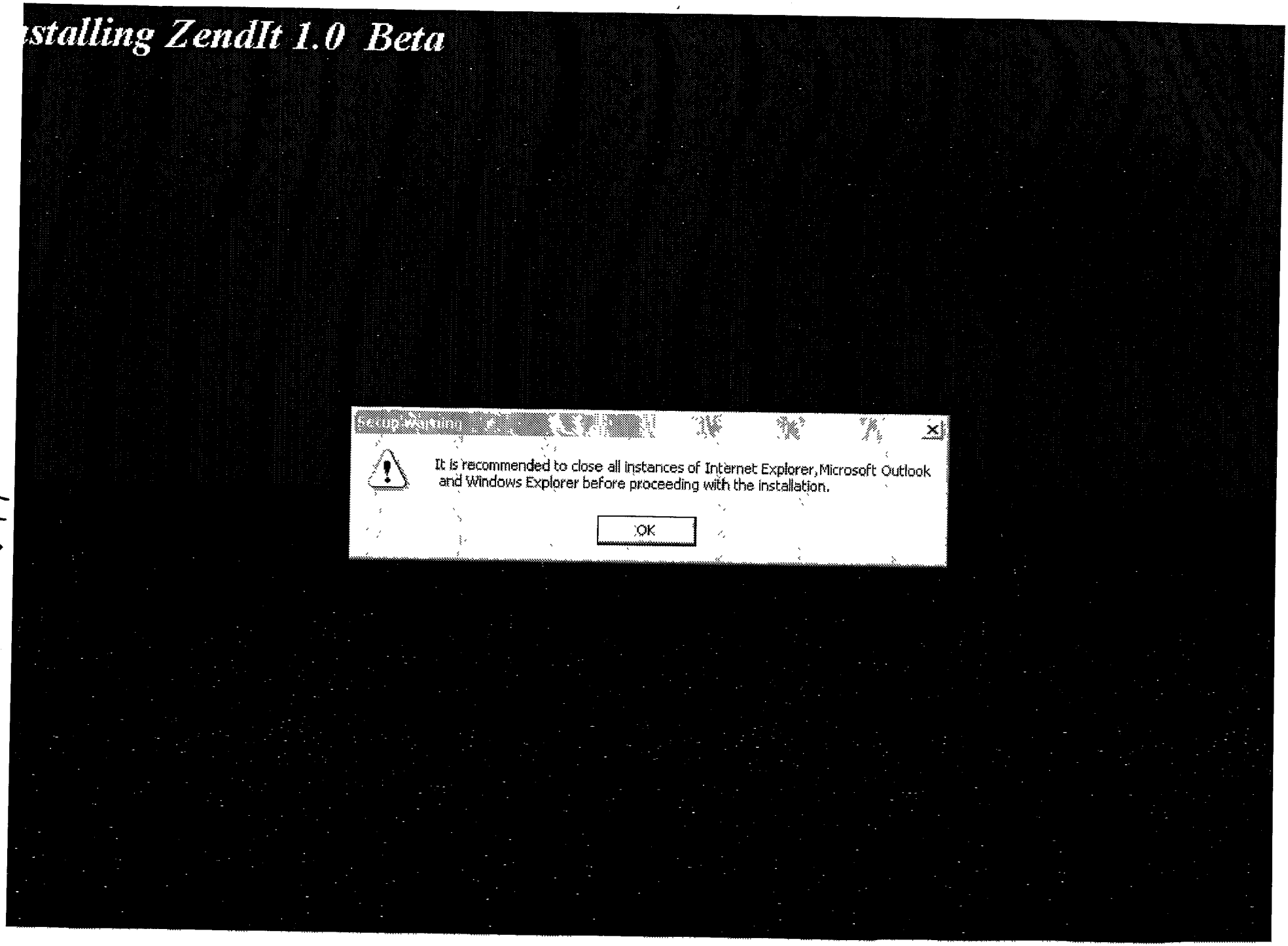


Logoff Zend DZend ZPanel Vault Help

xendit has logged on successfully

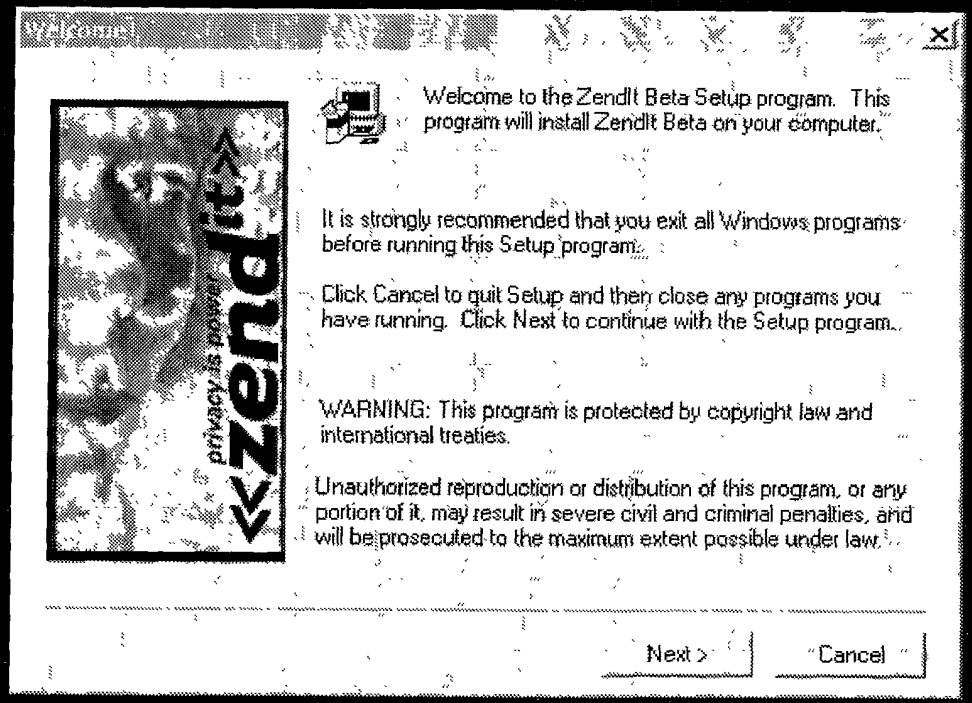
Internet

Installation

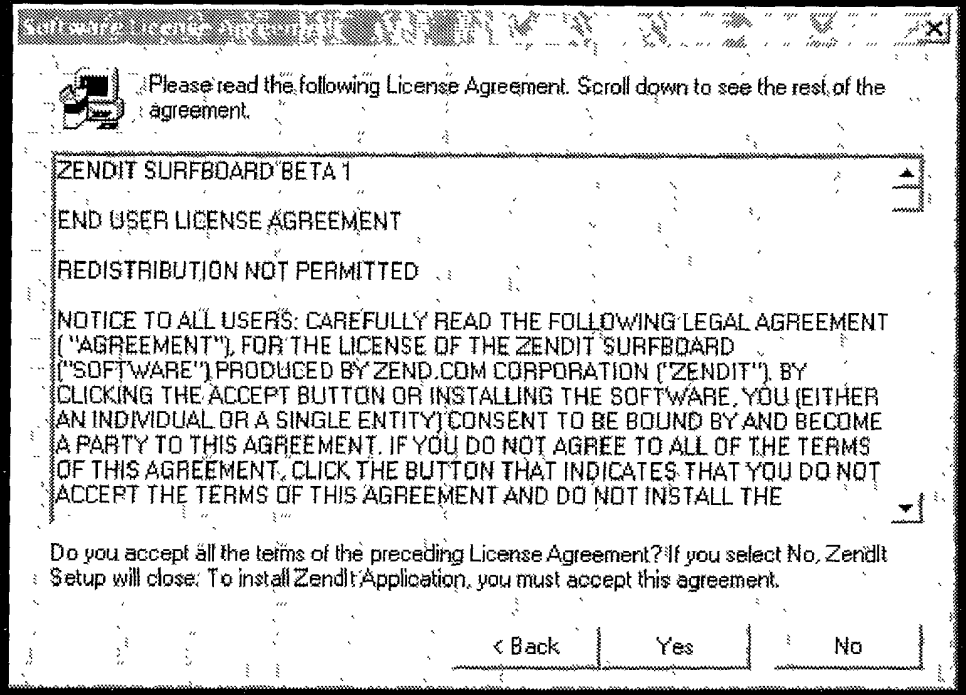

stalling ZendIt 1.0 Beta
Setup Warning
It is recommended to close all instances of Internet Explorer, Microsoft Outlook and Windows Explorer before proceeding with the installation.
OK

160

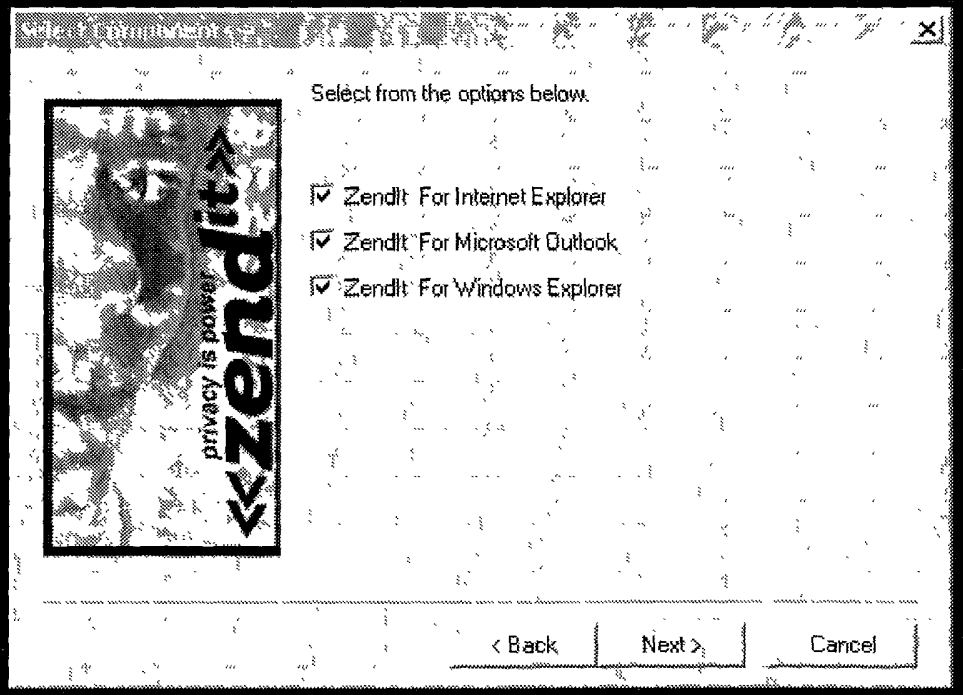
stalling ZendIt 1.0 Beta



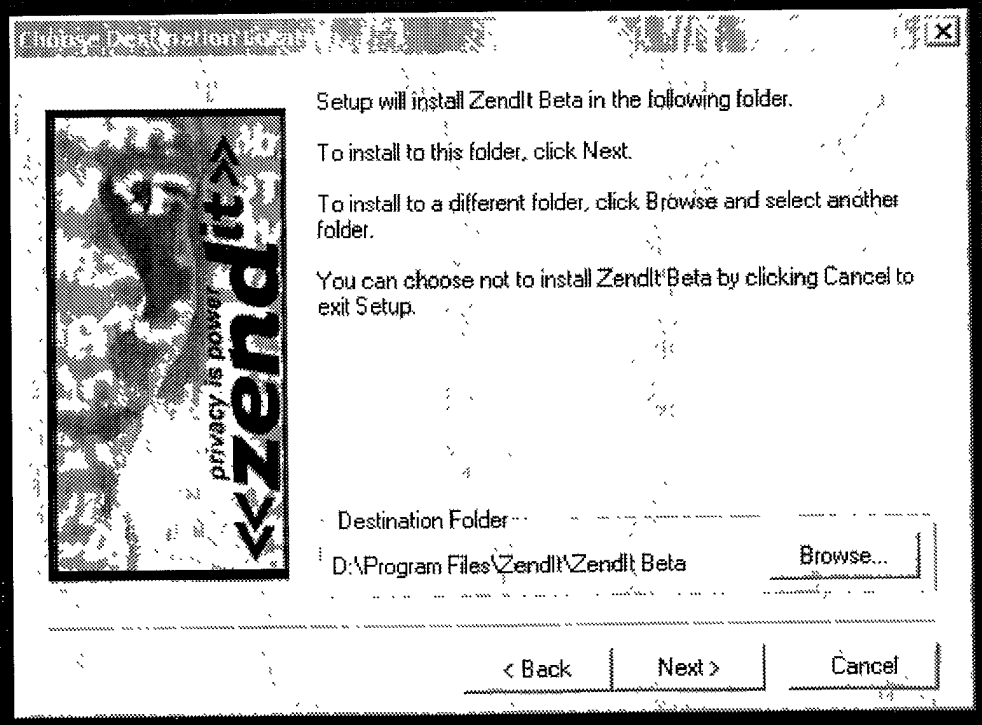
stalling ZendIt 1.0 Beta



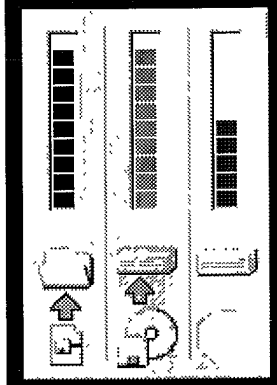
stalling ZendIt 1.0 Beta

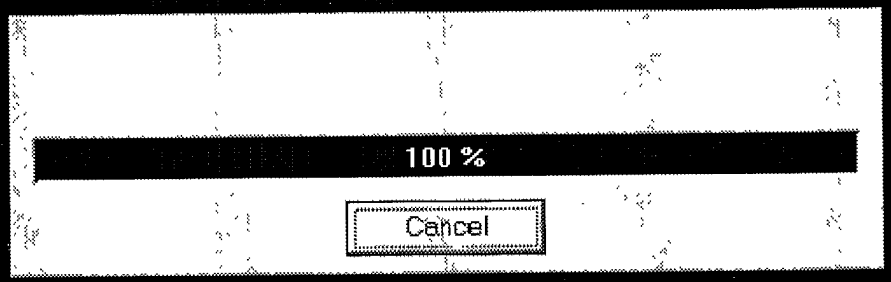


stalling ZendIt 1.0 Beta



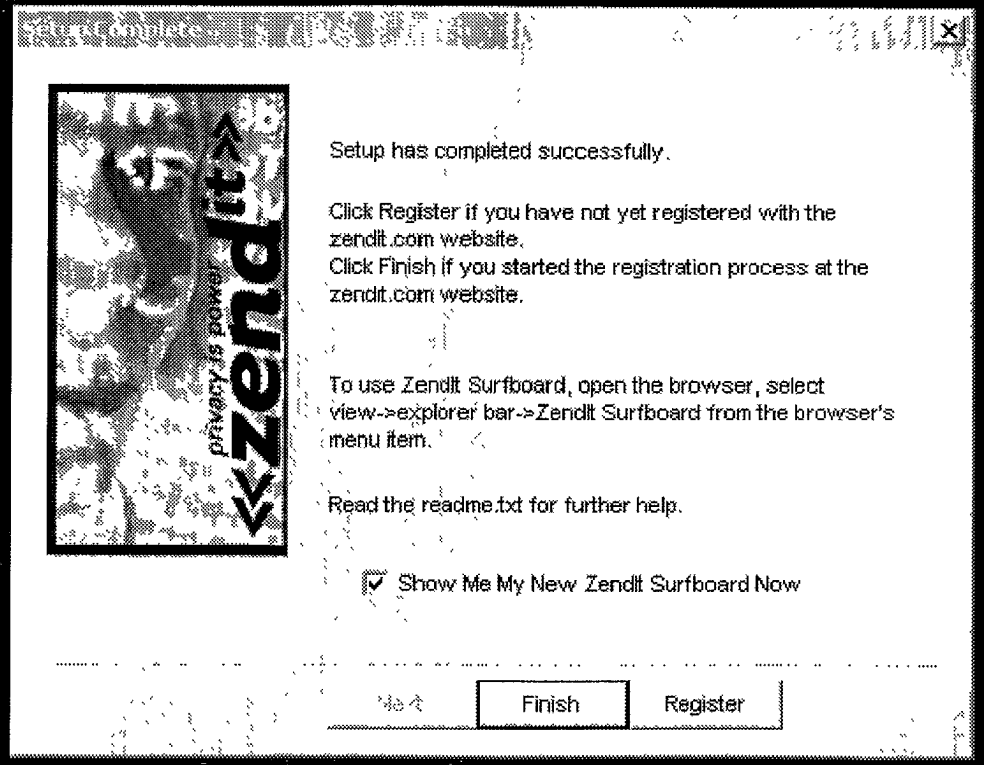
stalling ZendIt 1.0 Beta





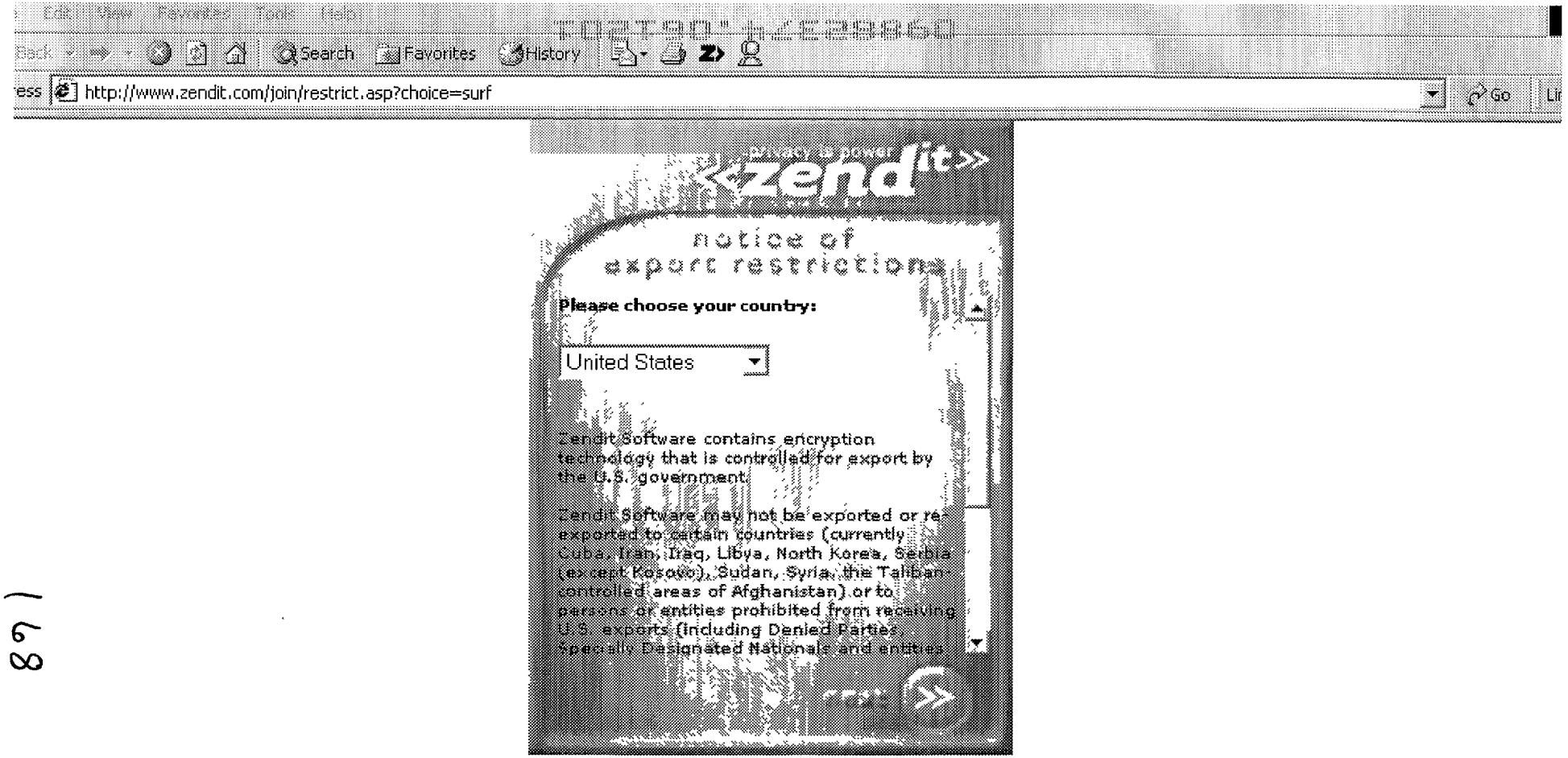
591

stalling ZendIt 1.0 Beta



9 9 1

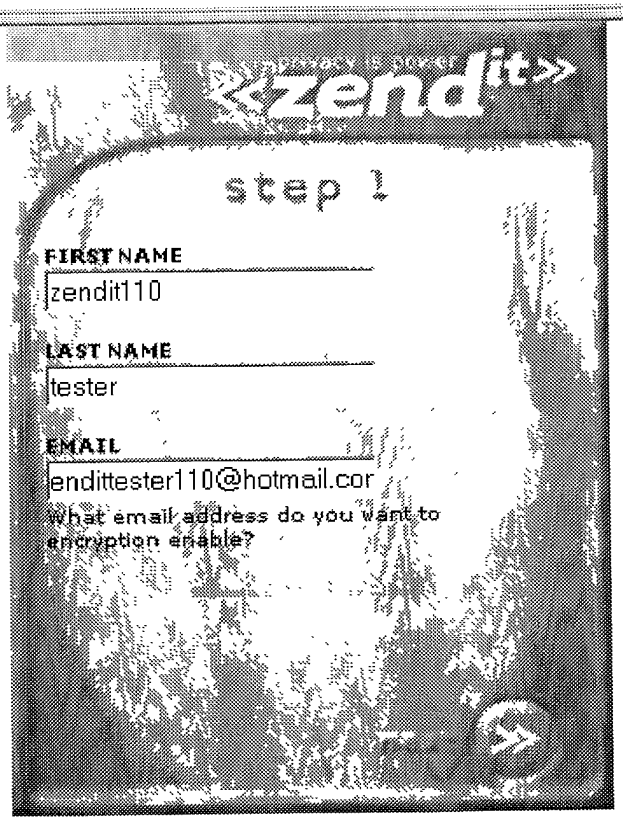
Surfboard Registration (new user)


Search
Favorites
History
http://www.zendit.com/join/restrict.asp?choice=surf
Go
zendit
notice of
export restrictions
Please choose your country:
United States
Zendit Software contains encryption technology that is controlled for export by the U.S. government.
Zendit Software may not be exported or re-exported to certain countries (currently Cuba, Iran, Iraq, Libya, North Korea, Serbia (except Kosovo), Sudan, Syria, the Taliban-controlled areas of Afghanistan) or to persons or entities prohibited from receiving U.S. exports (including Denied Parties, Specially Designated Nationals and entities
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Local intranet

8 of 1

Back Search Favorites History

ess http://www.zendit.com/join/signup1.asp?choice=surf Go

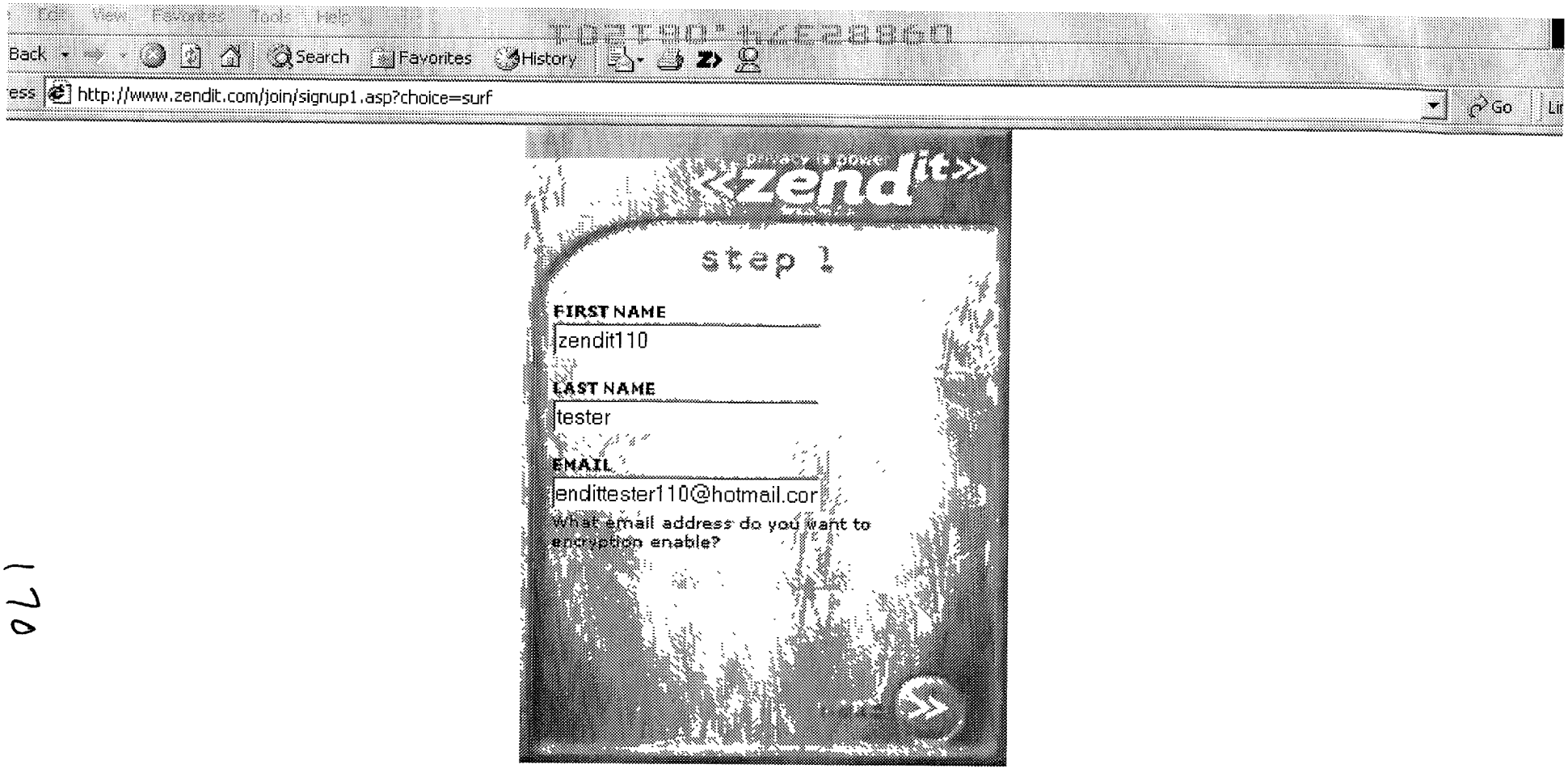


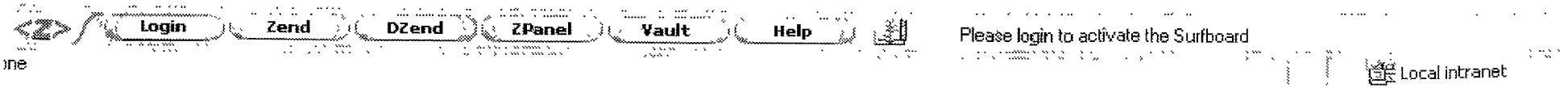
Login Zend DZend ZPanel Vault Help

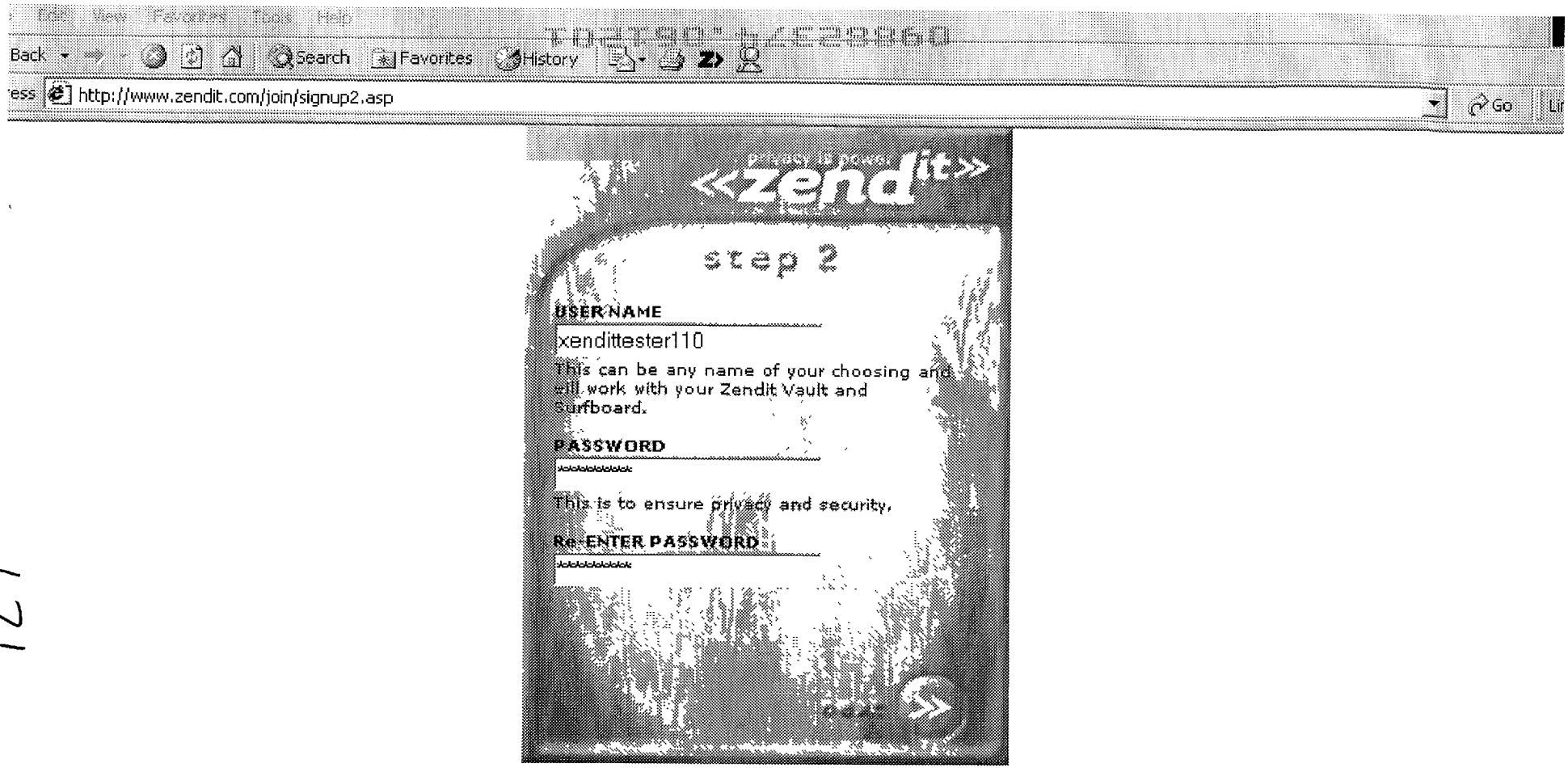
Please login to activate the Surfboard

Local intranet

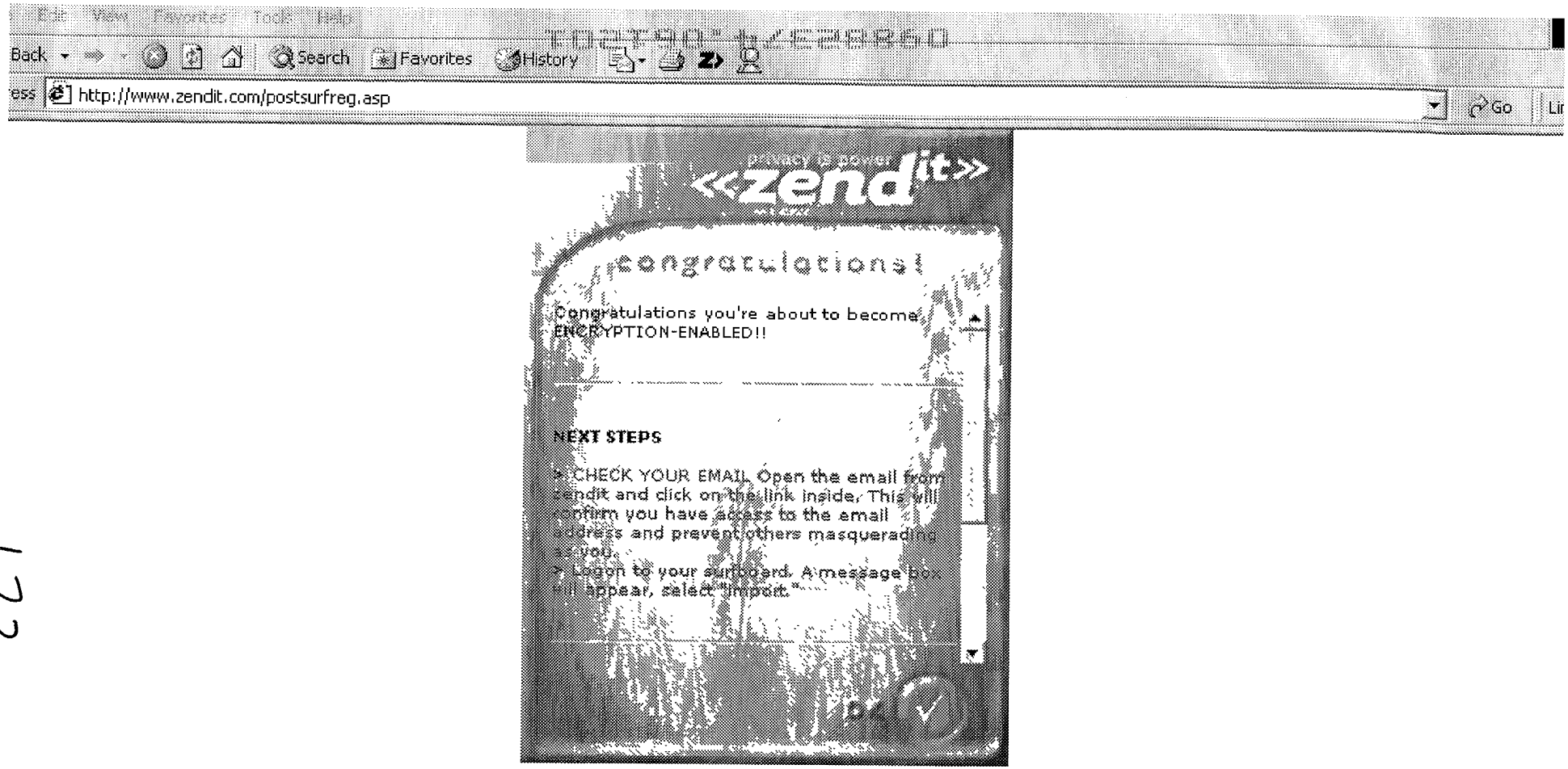
one


Back
Search
Favorites
History
http://www.zendit.com/join/signup1.asp?choice=surf
Go
zend it
step 1
FIRST NAME
zendit110
LAST NAME
tester
EMAIL
endittester110@hotmail.cor
what email address do you want to encryption enable?

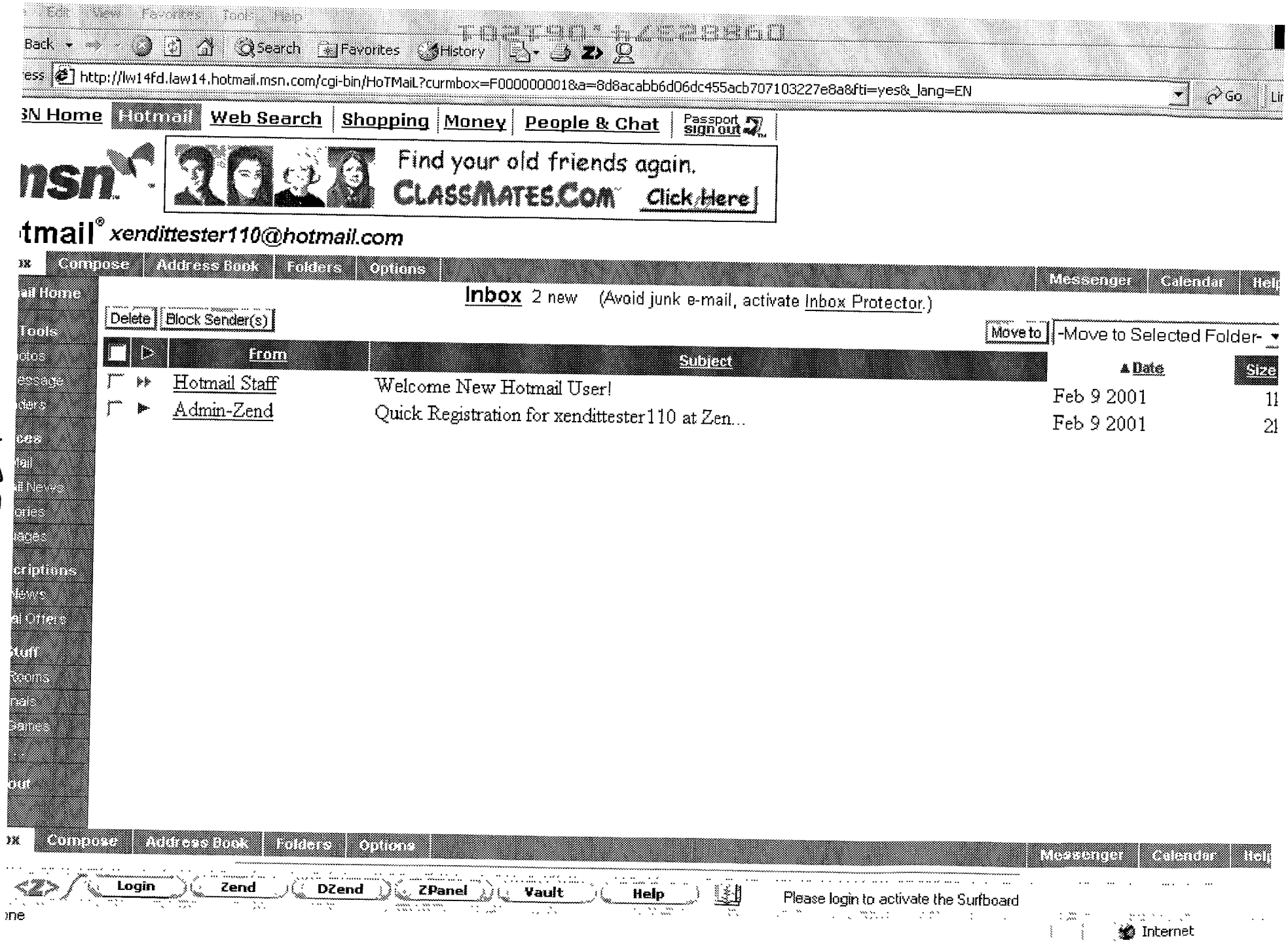

Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Local intranet

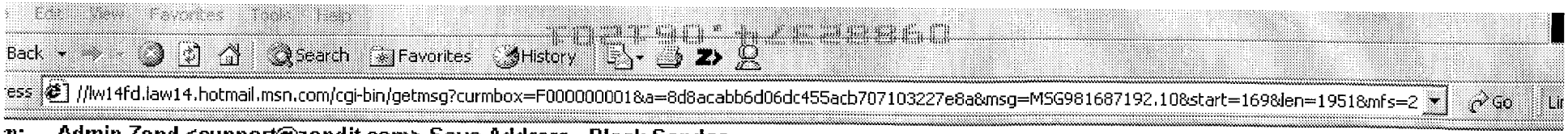

Back
Search
Favorites
History
http://www.zendit.com/join/signup2.asp
Go
zendit
step 2
USER NAME
xendittester110
This can be any name of your choosing and will work with your Zendit Vault and Surfboard.
PASSWORD
This is to ensure privacy and security.
Re-ENTER PASSWORD
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Local intranet

171


Back
Search
Favorites
History
http://www.zendit.com/postsurfreg.asp
Go
Congratulations!
Congratulations you're about to become ENCRYPTION-ENABLED!!
NEXT STEPS
1. CHECK YOUR EMAIL Open the email from zendit and click on the link inside. This will confirm you have access to the email address and prevent others masquerading as you.
2. Logon to your surfboard. A message box will appear, select "Import."
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Local intranet

171


Back
Search
Favorites
History
http://lw14fd.law14.hotmail.msn.com/cgi-bin/HoTMaiL?curmbox=F000000001&a=8d8acabb6d06dc455acb707103227e8a&fti=yes&_lang=EN
Go
MSN Home
Hotmail
Web Search
Shopping
Money
People & Chat
Passport sign out
msn
Find your old friends again.
CLASSMATES.COM
Click Here
tmail® xenditester110@hotmail.com
Compose
Address Book
Folders
Options
Messenger
Calendar
Inbox 2 new (Avoid junk e-mail, activate Inbox Protector.)
Delete
Block Sender(s)
Move to
-Move to Selected Folder-
From
Subject
Date
Size
Hotmail Staff
Welcome New Hotmail User!
Feb 9 2001
Admin-Zend
Quick Registration for xenditester110 at Zen...
Feb 9 2001
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
Internet



n: **Admin-Zend <support@zendit.com>** **Save Address** - **Block Sender**
"xendittester110@hotmail.com" <xendittester110@hotmail.com> **Save Address**
ject: Quick Registration for xendittester110 at ZendIt
Fri, 09 Feb 2001 02:49:13 (GMT)

Reply | Reply All | Forward | Delete | Previous | Next | Close

< < < Z E N D I T C O N F I R M A T I O N > > > > >

nk you for registering with Zendit!

C O N F I R M O W N E R S H I P
confirm your ownership of this email address please click on the link below:

p://www.zendit.com/quickconf.asp?LoginID=xendittester110&ConfID=PXWHTNBWXVSUBSPQIPUI&Email=xendittester110^hotmail.com

s is part of a confirmation process that protects you in case someone
empts to impersonate you and claims ownership of your account.

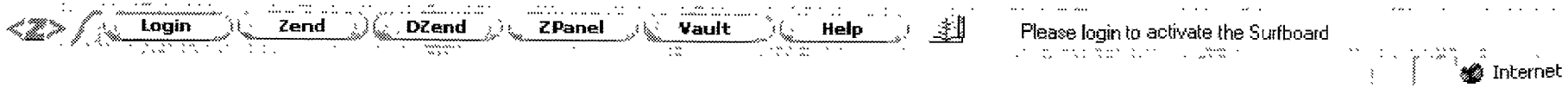
you did not register with Zendit, please do not confirm it.

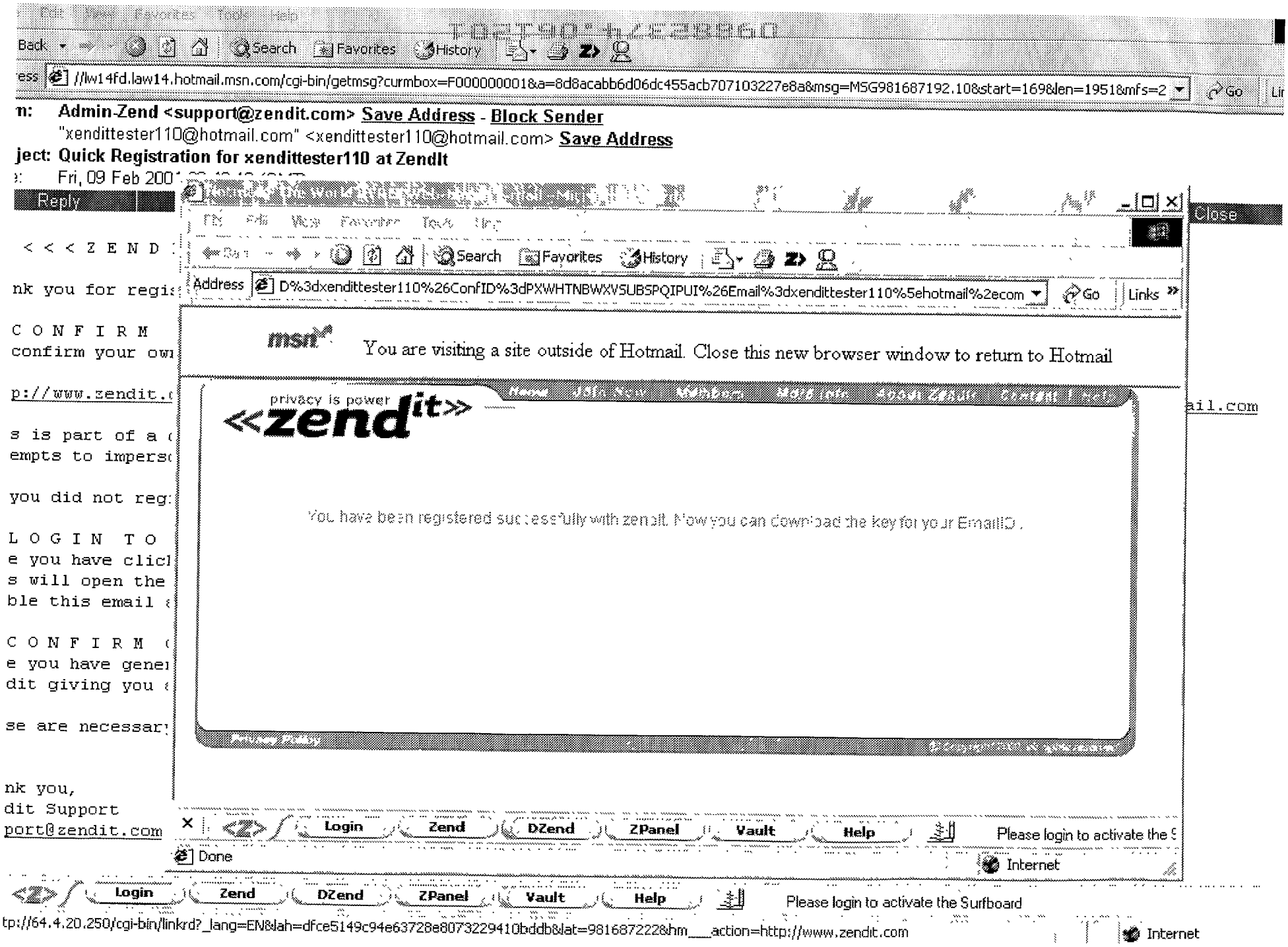
L O G I N T O S U R F B O A R D
e you have clicked on the above link, go ahead and login to the surfboard.
s will open the lock and key/certificate generator that will encryption
ble this email address.

C O N F I R M C E R T I F I C A T E
e you have generated your certificate you will receive one more email from
dit giving you a link to confirm your certificate.

se are necessary steps to ensure the integrity of your certificate.

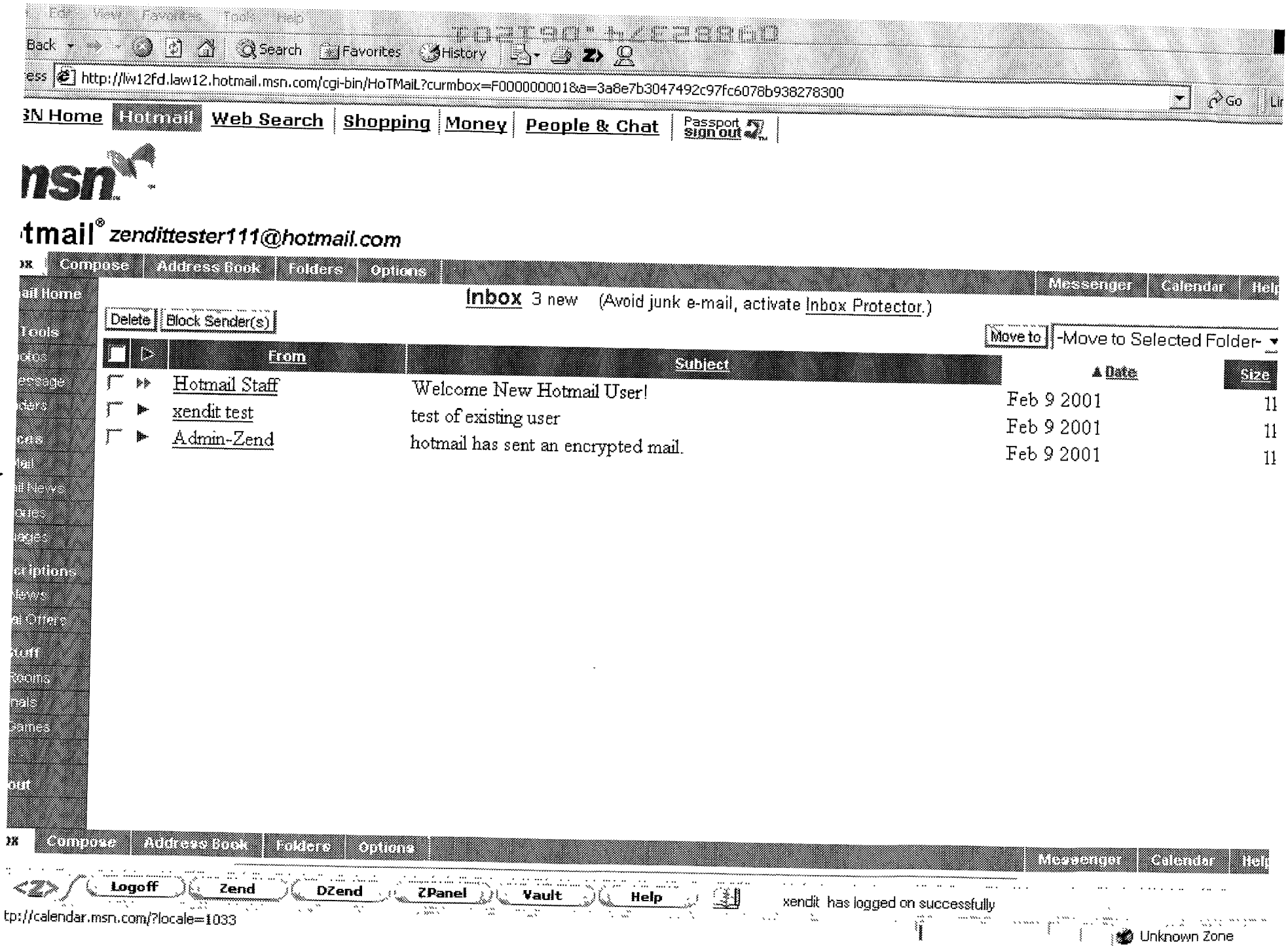
nk you,
dit Support
port@zendit.com



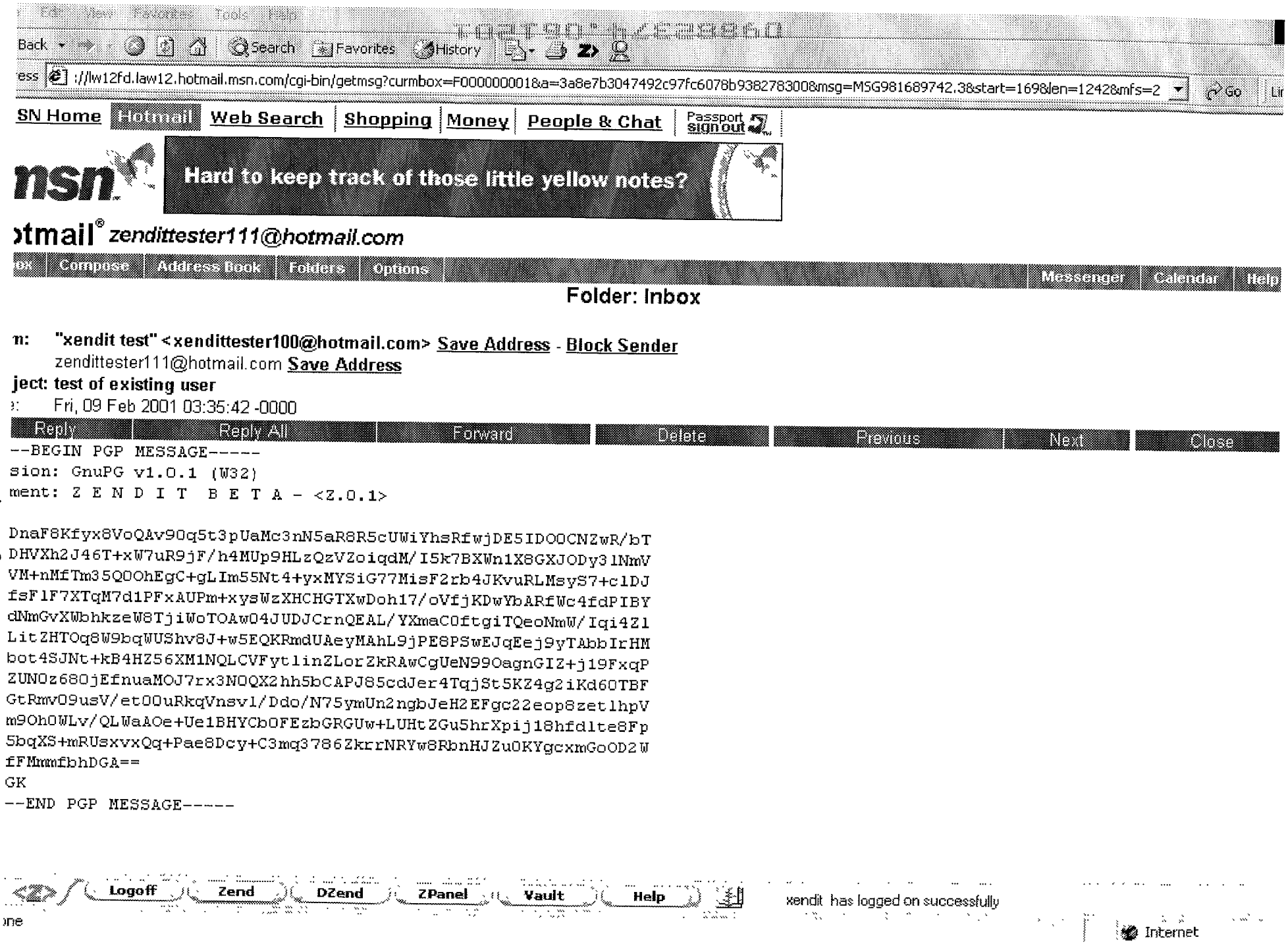

Back
Search
Favorites
History
ess //lw14fd.law14.hotmail.msn.com/cgi-bin/getmsg?curmbox=F000000001&a=8d8acabb6d06dc455acb707103227e8a&msg=MSG981687192.108&start=169&len=1951&mfs=2
Go
n: Admin-Zend <support@zendit.com> Save Address - Block Sender
"xendittester110@hotmail.com" <xendittester110@hotmail.com> Save Address
ject: Quick Registration for xendittester110 at ZendIt
: Fri, 09 Feb 200
Reply
Close
< < < Z E N D
nk you for regi
C O N F I R M
confirm your ow
p://www.zendit.
s is part of a
empts to impers
you did not reg
L O G I N T O
e you have clic
s will open the
ble this email
C O N F I R M
e you have gene
dit giving you
se are necessar
nk you,
dit Support
port@zendit.com
Address D%3dxendittester110%26ConfID%3dPXWHTNBWXVSUBSPQIPUI%26Email%3dxendittester110%5ehotmail%2ecom
Go
Links
msn
You are visiting a site outside of Hotmail. Close this new browser window to return to Hotmail
privacy is power
zendit
ail.com
You have been registered successfully with zendit. Now you can download the key for your EmailID .
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the S
Done
Internet
Login
Zend
DZend
ZPanel
Vault
Help
Please login to activate the Surfboard
tp://64.4.20.250/cgi-bin/linkrd?_lang=EN&lah=dfce5149c94e63728e8073229410bddb&lat=981687222&hm___action=http://www.zendit.com
Internet

175

Surfboard Registration (existing user)


Back
Search
Favorites
History
http://lw12fd.law12.hotmail.msn.com/cgi-bin/HoTMaiL?curmbox=F000000001&a=3a8e7b3047492c97fc6078b938278300
Go
SN Home
Hotmail
Web Search
Shopping
Money
People & Chat
Passport
sign out
msn
tmail® zenditester111@hotmail.com
Compose
Address Book
Folders
Options
Messenger
Calendar
Inbox 3 new (Avoid junk e-mail, activate Inbox Protector.)
Delete
Block Sender(s)
Move to
-Move to Selected Folder-
From
Subject
Date
Size
Hotmail Staff
Welcome New Hotmail User!
Feb 9 2001
11
xendit test
test of existing user
Feb 9 2001
11
Admin-Zend
hotmail has sent an encrypted mail.
Feb 9 2001
11
Logoff
Zend
DZend
ZPanel
Vault
Help
xendit has logged on successfully
tp://calendar.msn.com/?locale=1033
Unknown Zone

177



Back Search Favorites History

ess ://lw12fd.law12.hotmail.msn.com/cgi-bin/getmsg?curmbox=F000000001&a=3a8e7b3047492c97fc6078b938278300&msg=MSG981689742.3&start=1698&len=12428&mfs=2 Go

SN Home | Hotmail | Web Search | Shopping | Money | People & Chat | Passport sign out

Hard to keep track of those little yellow notes?

otmail® zendittester111@hotmail.com

pt | Compose | Address Book | Folders | Options | Messenger | Calendar | Help

Folder: Inbox

n: "xendit test" <xendittester100@hotmail.com> Save Address - Block Sender
zendittester111@hotmail.com Save Address
ject: test of existing user
: Fri, 09 Feb 2001 03:35:42 -0000

Reply | Reply All | Forward | Delete | Previous | Next | Close

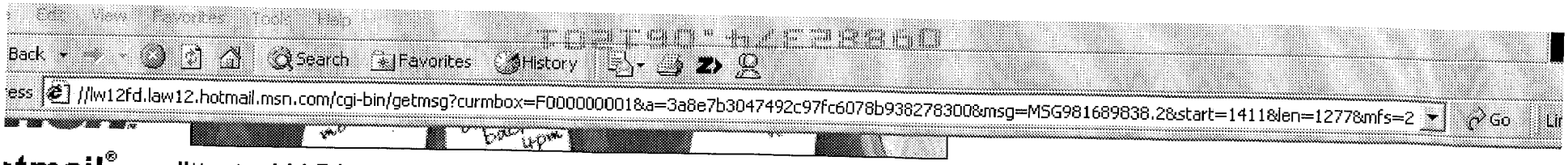
```
--BEGIN PGP MESSAGE-----
sion: GnuPG v1.0.1 (W32)
ment: Z E N D I T  B E T A - <Z.0.1>

DnaF8Kfyx8VoQAv90q5t3pUaMc3nN5aR8R5cUWiYhsRfwjDE5IDOOCNZwR/bT
DHVXh2J46T+xW7uR9jF/h4MUp9HLzQzVZoiqdM/I5k7BXWn1X8GXJODy3lNmV
VM+nMfTm35QOOhEgC+gLIm55Nt4+yxMYSiG77MisF2rb4JKvuRLMsyS7+clDJ
fsFlF7XTqM7d1PFxAUPm+xysWzXHCHGTXwDoh17/oVfjKDwYbARfWc4fdPIBY
dNmGvXWbhkzeW8TjiWoTOAwO4JUDJCrnQEAL/YXmaC0ftgiTQeoNmW/Iqi4Z1
LitZHTOq8W9bqWUShv8J+w5EQKRmdUAeyMAhL9jPE8PSwEJqEej9yTAbbIrHM
bot4SJNt+kB4HZ56XM1NQLCVFytlinZLorZkRAwCgUeN99OagnGIZ+j19FxqP
ZUNOz68OjEfnuaMOJ7rx3NOQX2hh5bCAPJ85cdJer4TqjSt5KZ4g2iKd60TBF
GtRmvO9usV/et0OuRkqVnsvl/Ddo/N75ymUn2ngbJeH2EFgc22eop8zetlhpV
m9OhOWLv/QLWaAOe+Ue1BHYCbOFEzbGRGUw+LUHtZGu5hrXpij18hfdlte8Fp
5bqXS+mRUsxvxQq+Pae8Dcy+C3mq3786ZkrrNRYw8RbnHJZuOKYgcxmGoOD2W
fFMmmfbhDGA==
GK
--END PGP MESSAGE-----
```

Logoff | Zend | DZend | ZPanel | Vault | Help

xendit has logged on successfully

one

Internet



tmail® *zendittester111@hotmail.com*

Compose | Address Book | Folders | Options | Messenger | Calendar | Help

Folder: Inbox

n: **Admin-Zend <support@zendit.com>** **Save Address** - **Block Sender**
<zendittester111@hotmail.com> **Save Address**
ject: hotmail has sent an encrypted mail.
: Fri, 09 Feb 2001 03:32:53 (GMT)

Reply | Reply All | Forward | Delete | Previous | Next | Close

mail, recently sent you a privileged and confidential email. If you want to
rypt it, please click on the following link to confirm this is YOUR email
ress.

s confirmation protects you in case someone tries to forge ownership of YOUR
il address.

p://www.zendit.com/default.asp?EmailID=zendittester111^hotmail.com&ConfID=JVHQUVMDQYSYJLZMULHW&tempconf=1

nk you,
dit Support
port@zendit.com

vacy is Power

Reply | Reply All | Forward | Delete | Previous | Next | Close

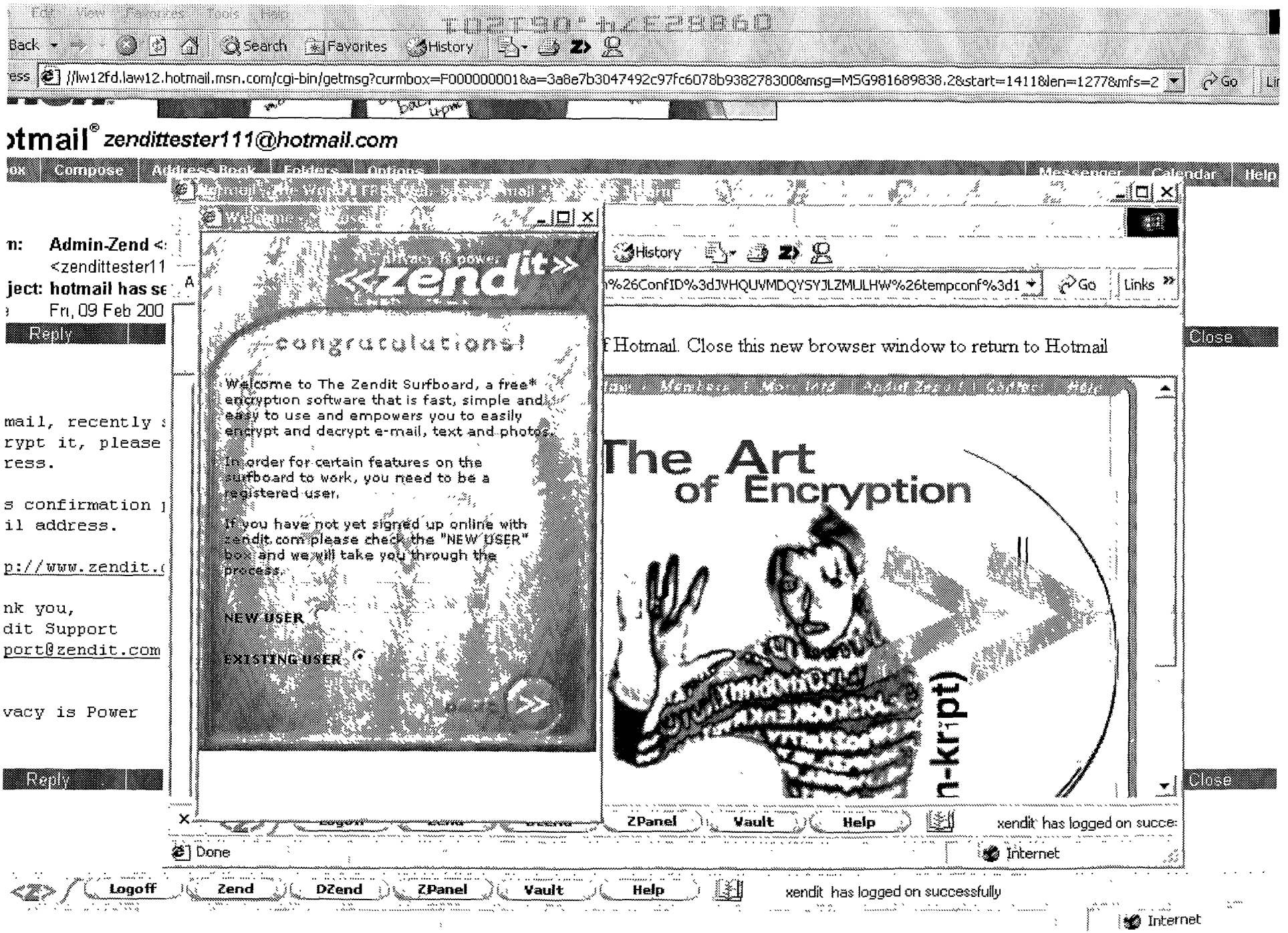
Move To (Move to Selected Folder)

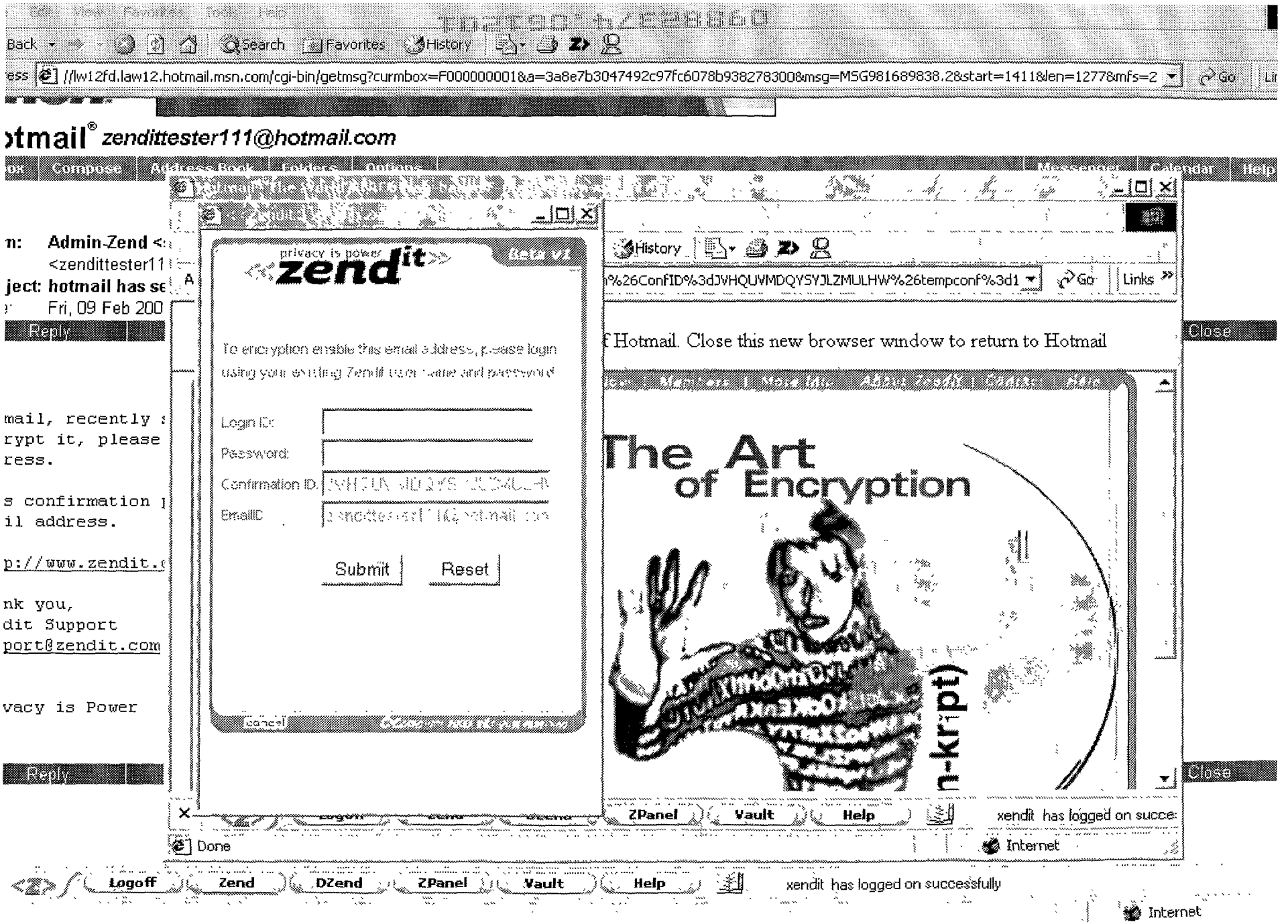
Logoff | Zend | DZend | ZPanel | Vault | Help

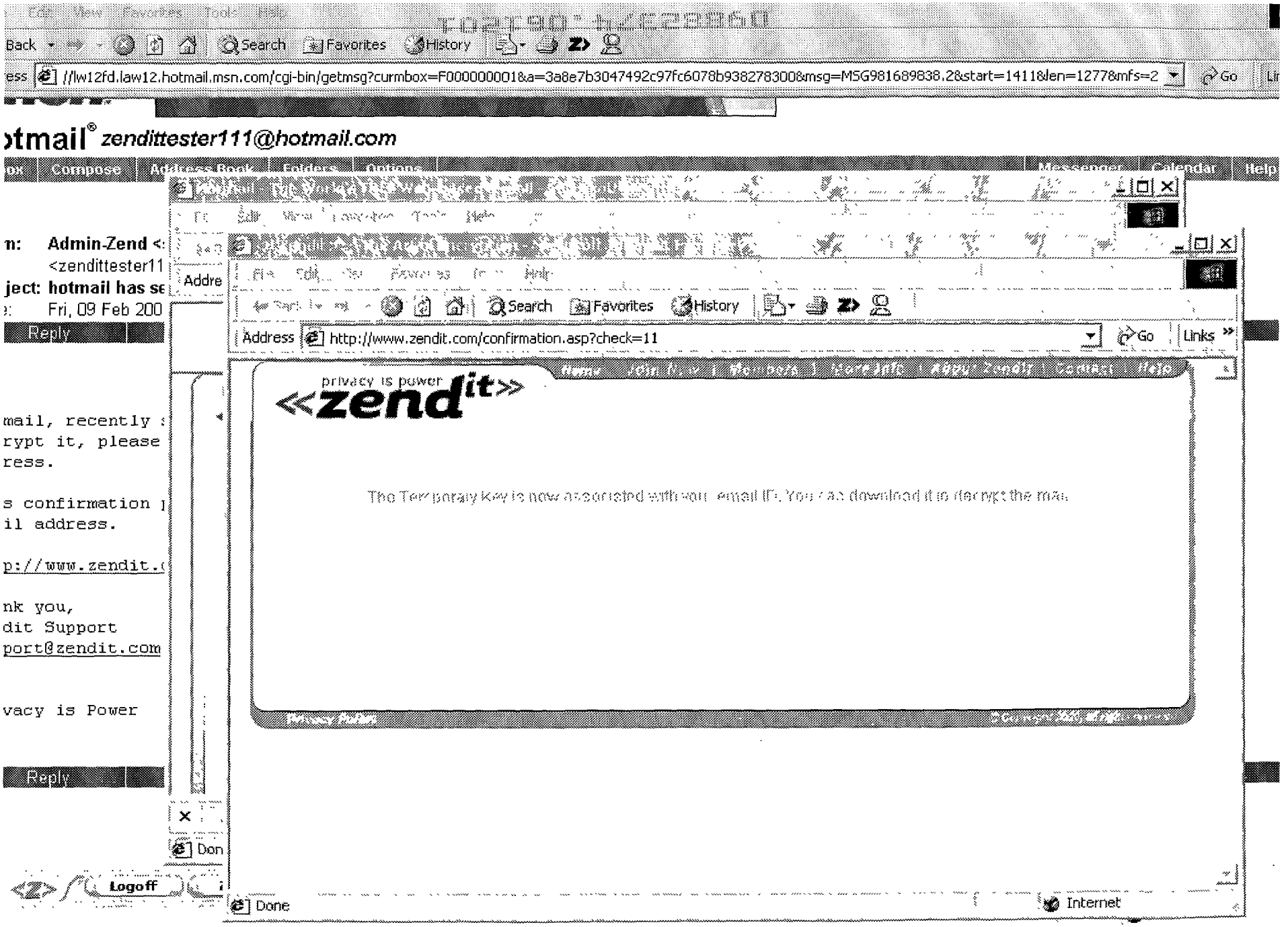
xendit has logged on successfully

one

Internet


Back
Search
Favorites
History
//lw12fd.law12.hotmail.msn.com/cgi-bin/getmsg?curmbox=F000000001&a=3a8e7b3047492c97fc6078b9382783008msg=MSG981689838.28start=1411&len=1277&mfs=2
Go
otmail® zenditester111@hotmail.com
Compose
Calendar
Help
Admin-Zend <
<zenditester11
ject: hotmail has se
Fri, 09 Feb 200
Reply
Close
mail, recently
rypt it, please
ress.
s confirmation
il address.
p://www.zendit.
nk you,
dit Support
port@zendit.com
vacy is Power
Reply
Close
Welcome to Zendit
Privacy is power
<zendit>>
congratulations!
Welcome to The Zendit Surfboard, a free* encryption software that is fast, simple and easy to use and empowers you to easily encrypt and decrypt e-mail, text and photos.
In order for certain features on the surfboard to work, you need to be a registered user.
If you have not yet signed up online with zendit.com please check the "NEW USER" box and we will take you through the process.
NEW USER
EXISTING USER
History
Go
Links
f Hotmail. Close this new browser window to return to Hotmail
The Art of Encryption
ZPanel
Vault
Help
xendit has logged on succe
Done
Internet
Logoff
Zend
DZend
ZPanel
Vault
Help
xendit has logged on successfully
Internet


Back
Search
Favorites
History
//lw12fd.law12.hotmail.msn.com/cgi-bin/getmsg?curmbox=F000000001&a=3a8e7b3047492c97fc6078b938278300&msg=M5G981689838.2&start=1411&len=1278&mfs=2
Go
tmail® zendittester111@hotmail.com
Compose
Address Book
Calendar
Help
n: Admin-Zend <
<zendittester11
ject: hotmail has se
Fri, 09 Feb 200
Reply
mail, recently
rypt it, please
ress.
s confirmation
il address.
p://www.zendit.
nk you,
dit Support
port@zendit.com
vacy is Power
Reply
privacy is power
zendit
To encryption enable this email address, please login
using your existing Zendit user name and password
Login ID:
Password:
Confirmation ID:
EmailID:
Submit
Reset
Cancel
History
Go
Links
Hotmail. Close this new browser window to return to Hotmail
Close
The Art
of Encryption
n-krypt)
ZPanel
Vault
Help
xendit has logged on succe
Done
Internet
Logoff
Zend
DZend
ZPanel
Vault
Help
xendit has logged on successfully
Internet


Back
Search
Favorites
History
Go
//lw12fd.law12.hotmail.msn.com/cgi-bin/getmsg?curmbox=F000000001&a=3a8e7b3047492c97fc6078b938278300&msg=MSG981689838.2&start=1411&len=1277&mfs=2
tmail® zendittester111@hotmail.com
Compose
Reply
Admin-Zend
hotmail has se
Fri, 09 Feb 200
mail, recently
rypt it, please
ress.
s confirmation
il address.
p://www.zendit.
nk you,
dit Support
port@zendit.com
vacy is Power
Logoff
Address
http://www.zendit.com/confirmation.asp?check=11
Links
privacy is power
zendit
Done
Internet

United States Patent & Trademark Office

Office of Initial Patent Examination -- Scanning Division

Application deficiencies found during scanning:

☐ Page(s)_____of_____ were not present for scanning. (Document title)

☐ Page(s)_____of_____were not present for scanning. (Document title)

☑ ***Scanned copy is best available.*** Drawings.

Miscellaneous.